DM 14,- Ös 120,- sfr 14,-

SONDERHEFT
- Reviews, Spiele, Anwendungen -

CPC Schneider INTERNATIONAL

Sonderheft
3/86

Listings:

FLUGSIMULATOR
GANGSTER
THE DESIGNER
ROLLENSPIEL-ADVENTURE

Spiele:

DIE CREME DER CPC-SPIELE

Spieletips:

KARTEN & LÖSUNGSHILFEN

CPC + Joyce:

3D-GRAFIK IN LOGO

Hardware:

ECHTZEITUHR IM SELBSTBAU
RS 232-SCHNITTSTELLE

Der neue Schneider PC in Wort und Bild

INFOCOM™

Kommt!

Direkt und ohne Umwege. Nach Deutschland.

Und bringt den kompletten Katalog mit.

Die Legenden – die Klassiker – die Neuheiten

FÜR DEN SCHNEIDER CPC

DON'T PANIC!



INFOCOM c/o Activision Deutschland GmbH Postfach 760680 · 2000 Hamburg 76

Impressum

Herausgeber
Christian Widuch

Chefredakteur
Stefan Ritter

Redaktion
Thomas Morgen (TM), Heinrich Stiller (HS), Michael Ceol (MC), Michael Ebbrecht (ME)

Satz
Silvia Führer, Renate Wells, Claudia Küllmer

Gestaltung
Renate Wells, Gerd Köberich, Yvonne Tischendorf

Grafik/Illustration
Heinrich Stiller

Fotografie
Gerd Köberich

Anzeigenleitung
Wolfgang Schnell
Es gilt die Anzeigenpreisliste Nr. 2 vom 1.1.1986

Freie Mitarbeiter
K.J. Wolf, M. Kneip, T. Kuhn, M.+A. Pisters, T. Kochmann, R. Backes, R. Krampfel, B. Morell, J. Schweda, T. Fippl, H. Ziegler, W. Wantia, M. Uphoff, R. Bodinger, T. Schulte, O. Heggelbacher, M. Wegmann, P. Benes, W. Tebes, A. Stolzenberger, Dipl.-Ing. H. Bruhns, S. Mayer, H. Müller R. Alff

Anschrift Verlag/Redaktion
DMV Daten & Medien Verlagsges. mbH
Postfach 250
3440 Eschwege
Telefon: 05651/8702

Vertrieb
Verlagsunion
Friedrich-Bergius-Straße 20
6200 Wiesbaden

Druck
Druckerei Jungfer, 3420 Herzberg

Bezugspreise
Schneider CPC International Sonderheft 3/86 erscheint einmal jährlich.

Einzelpreis DM 14,-/sfr .14,-/ÖS 120,-



Liebe Leser!

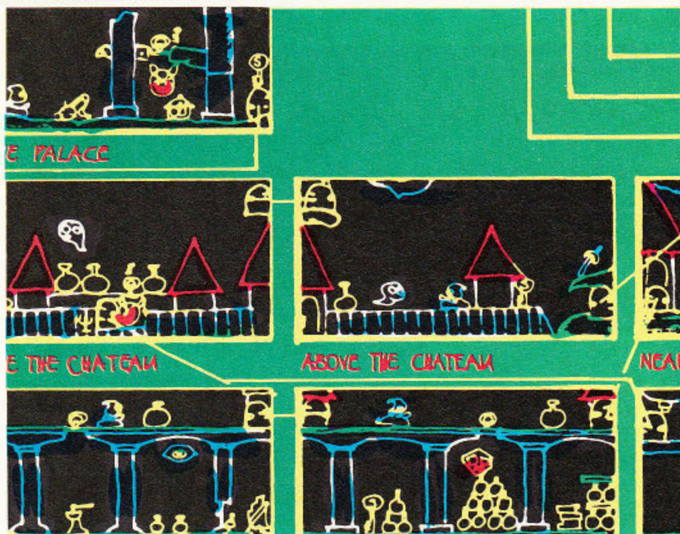
Das Jahr 1986 hat seinen Zenith überschritten. Obgleich es noch nicht ganz vorbei ist, ein guter Grund, sich das Schneider-Jahr einmal genauer zu betrachten. Dies wollen wir, zusammen mit Ihnen, zumindest im Bereich der Spielsoftware tun. Wir haben noch einmal die Software des Jahres 1986 durchgesehen und einen repräsentativen Querschnitt zusammengefaßt. Kurz und prägnant besprochen, mit einem Bildschirmfoto versehen, ist daraus fast so etwas wie eine Chronik mit Einkaufsführerqualitäten geworden. Diese Sammlung wird Ihnen sicherlich nicht zuletzt bei Ihren Weihnachtseinkäufen gute Dienste leisten.

Außerdem finden Sie in unserem dritten Sonderheft einige unterhaltsame und nützliche Listings. Erstmals ist es uns hier gelungen, mit dem babylonischen Basic-Gewirr auf den CPC's aufzuräumen. Ein Emulator macht es möglich. Mit diesem Emulator, der übrigens auch von unserem Context-Autor Matthias Uphoff geschrieben wurde, ist es möglich, auf dem CPC 464 Programme mit den Basic-Befehlen des CPC 664 und des 6128 laufen zu lassen. Der Emulator simuliert diese Befehle softwaremäßig.
Dies erklärt auch, daß bei unseren Listings die 464, 664, 6128 Markierungen wegfielen. Sie werden einfach nicht mehr gebraucht.

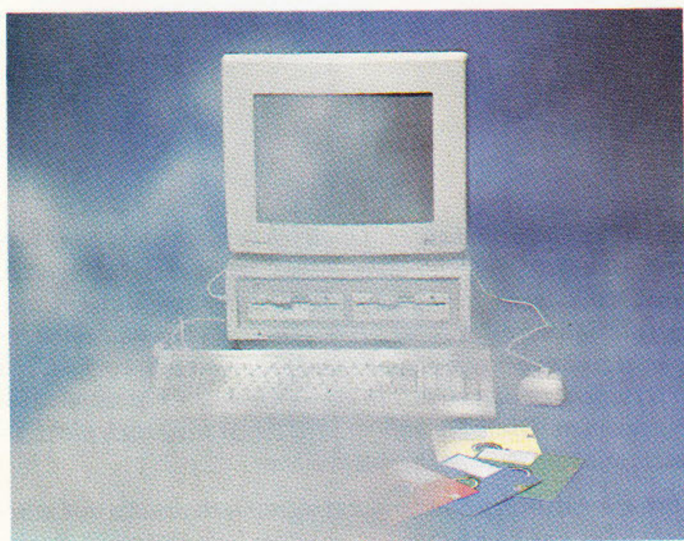
Wir hoffen, daß es uns gelungen ist, auch in unserem dritten Sonderheft eine Mischung zu finden, die Ihnen, liebe Leser, dabei hilft, das Beste aus Ihrem Rechner herauszuholen, oder aber das Interessanteste für Ihren Rechner finden.

Heinrich Stiller

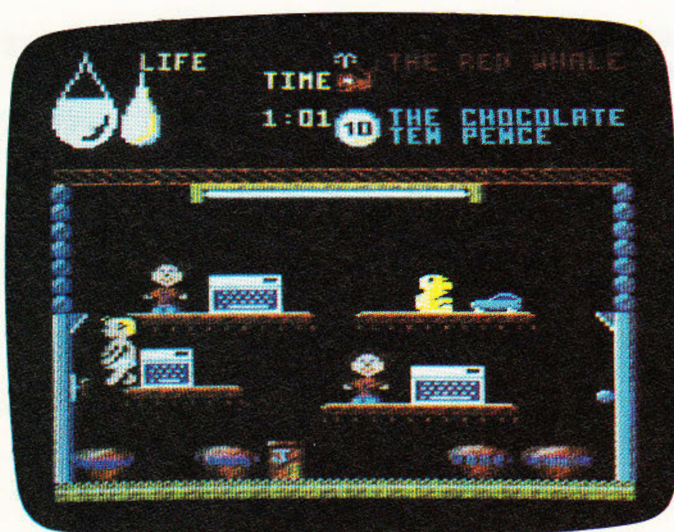
H. Stiller



Unter den vielen interessanten Spielekarten finden Sie auch den definitiven Sorcery Plus-Lageplan. Sicherlich eine große Hilfe bei so manchem heißen Spiel.



Das Geheimnis ist gelüftet! Schneiders neuer PC präsentiert sich der wartenden Öffentlichkeit. In Wort und Bild vorgestellt auf Seite 130.



Micro Gens Wally Week-Adventures haben sich einen festen Platz auf dem Softwaremarkt erkämpft. Wie Sie bei diesen Spielen auf einen grünen Zweig kommen, verrät das „Wally Week Quartett".

Artikel:

Spielekarten:

Spielelistings:

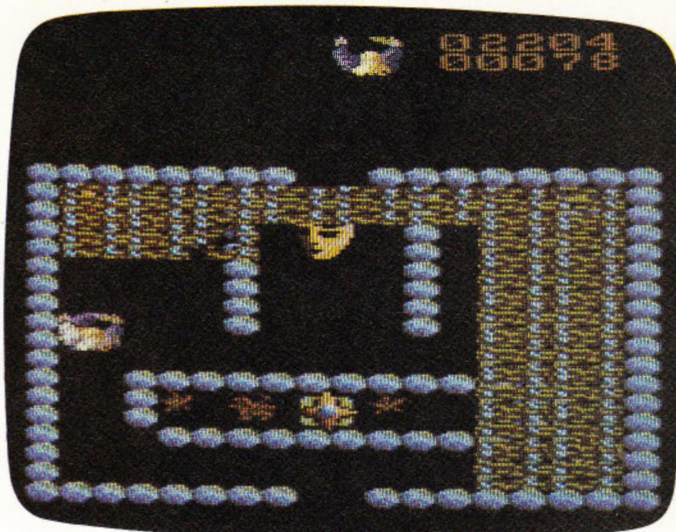
Hardware:

Anwenderprogramme :

Tips + Tricks:

Spieletips:

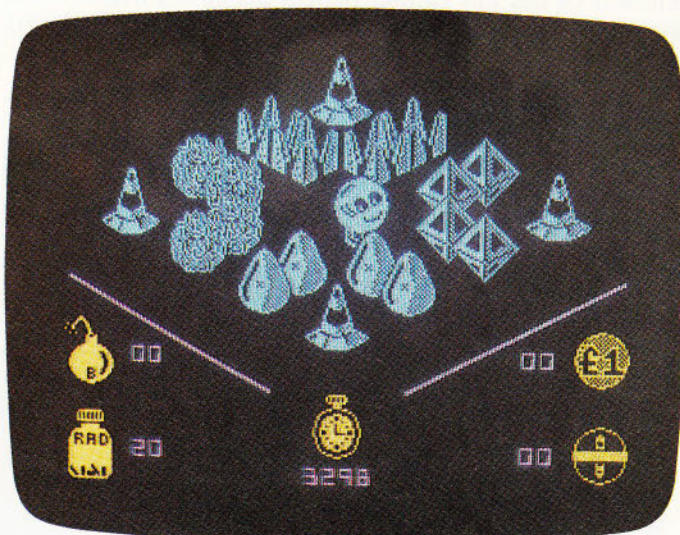
Rubriken :



Software satt. Im großen Review-Teil finden Sie eine Übersicht des Gamewaremarktes 1986.



Sucht man nach Pionieren unter den Abenteuersoftware-Herstellern, dauert es nicht lange, bis einem der Name „Infocom" begegnet. Was nun wirklich hinter diesem Namen steckt, erfahren Sie auf Seite 20.



Die Billigspiele sind eine nicht zu unterschätzende Komponente des Softwaremarktes geworden. Was sich dort getan hat, lesen Sie auf Seite 32.

Gameware '86

1986 war für den Spielefreund, der im Besitz eines Schneider Rechners ist, ein ausgesprochen ergiebiges Jahr. Dem CPC ist es gelungen, auch weiterhin in der Gunst der Anwender zu steigen. Durch eine weite Verbreitung des Rechners sahen auch die Softwarehäuser einen Markt wachsen, auf dem sie, wollten sie den Anschluß nicht verlieren, präsent sein mußten. Da der Schneider-User sich als kritischer Kunde erwies, zeigte sich, daß man sich mit minderwertigen Produkten auf diesem Markt nicht behaupten konnte. Der harte Konkurrenzkampf bescherte den Schneider-Anwendern eine wahre Flut qualitativ hochwertiger Software.

Ganz besonders im Bereich der Spiele ist auf diese Art und Weise so manches Kleinod angewandter Maschinenprogrammierung entstanden. Zwar ist es richtig, daß der Softwaremarkt mitunter genauso schnellebig ist wie die Musikbranche, aber auch hier gilt, daß ältere Produkte dem Spieler, der sie noch nicht kennt, bisweilen mehr Spaß und Unterhaltung bescheren, als etwa ein momentan mit viel Elan beworbenes Spiel.

Aber wie erkennt man nun, was man von den einzelnen Titeln halten soll, die auf dem Markt erhältlich sind? Information tut not.

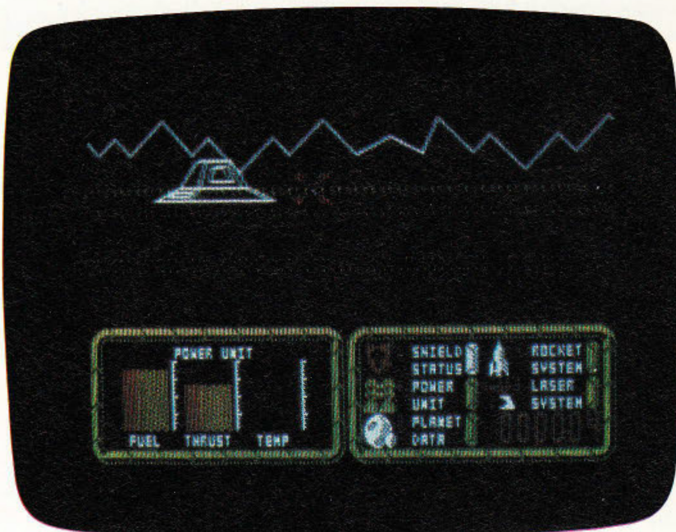
Nachfolgend an diese kleine Einleitung finden Sie einen Querschnitt durch das Spielejahr 1986. Wir haben aus unseren Softwarereviews ein repräsentatives Sortiment Software zusammengestellt. Prägnant und kritisch kommentiert, mit jeweils einem Bildschirmphoto versehen, stellen wir Ihnen hier die Entwicklungen eines Jahres vor.

Shoot em up

BATTLE BEYOND THE STARS: Nachdem die Space Invaders erst einmal das Laufen und Schießen gelernt hatten, war es nur ein kleiner Schritt bis zum Fliegen. Und wirklich - schon die nächste Generation hatte es gelernt;



Battle beyond the Stars



Battle of the Planets

abgesehen davon, waren sie inzwischen bunt geworden und nannten sich fortan "Galaxions.

Dieses Spiel war es dann auch, das den Autor von BATTLE mit ziemlicher Sicherheit beeinflußt hat. Es geht wieder einmal um die Verteidigung unseres Heimatplaneten gegen schießwütige Invasoren aus der siebten Galaxie. In wahren Kaskaden stürzen sie sich auf das am unteren Bildschirmrand hin und her fahrende Raumschiff des Spielers und machen dabei die tollsten Kapriolen. Grafisch und soundmäßig ist "Battle" sicherlich kein Highlight. Trotzdem ist es ein Spiel, das nicht zuletzt ob seines günstigen Preises, ein verhältnismäßig gutes Preis-/ Leistungsverhältnis aufweist und eigentlich eine ganze Menge Spaß bringen kann. Eines sei noch bemerkt: ein Kinderspiel ist der Kampf gegen die Aliens nicht gerade. Einiges an Übung will schon investiert sein, um hier einen anständigen Score zu erzielen.

BATTLE OF THE PLANETS: Auch dieses, von der englischen Firma Micro Gen realisierte Programm, entführt den Spieler in ein intergalaktisches Schlachtfeld. BOTP präsentiert sich in einem beachtlichen Gewand auf dem Bildschirm. Klar gezeichnete und perfekt animierte Vectorgrafiken versetzen den Spieler hautnah in das Spielgeschehen. Zwar ist der Sound ein wenig dürftig ausgefallen, aber dafür entschädigt eine umso komplexere Spielhandlung. Denn für ein Ballerspiel wird hier vom Spieler doch einiges an Strategie gefordert.

Da die Invasoren sich nicht einfach damit begnügen, alle im Weltraum befindlichen gegnerischen Raumschiffe zu vernichten, sondern auch noch spezielle Landefahrzeuge auf den einzelnen Planeten absetzen, muß immer abgewägt werden, ob man sich zuerst den Feinden im Weltraum entgegenstellt oder versucht, die drohende Vernichtung einzelner Planeten zu verhindern. Zwar ist auch hier ein gewisses Maß an Geschicklichkeit vonnöten, aber die Möglichkeit, das eigene Raumschiff immer wieder reparieren zu lassen, gibt dem Spieler Zeit mit den an Ihn gestellten Anforderungen zu wachsen.

COMMANDO: Diese langerwartete Umsetzung eines recht beliebten Arcaden-Automatenspiels durch Elite Systems kam in Deutschland mit einer kleinen Änderung auf den Markt. Anstelle der Soldaten aus Fleisch und Blut die der Spielfigur in der englischen Version das Leben



Commando

schwer machen, wurden für den deutschen Markt kleine silberne Roboter eingesetzt – wohl um einer drohenden Indizierung zu entgehen. Zwar wurde der Name auch geändert – in SPACE INVASION ,aber das interessierte niemanden. Die große Besonderheit bei diesem Spiel ist die Gefahr anatomischer Deformationen, durch exzessives Ballern.

Ein einzelner Super-Rambo, gesteuert vom Spieler, wird da von einer ganzen Armee angegriffen .Er muß nun versuchen, an die verschiedenen Basislager der angreifenden Truppen heranzukommen und diese zu zerstören. Auf dem Bildschirm sieht das so aus:

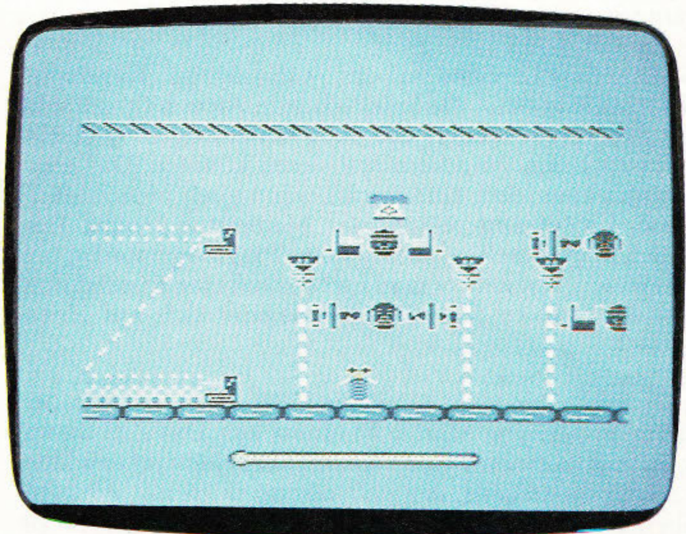
Die Figur des Helden jagt von unten nach oben durch ein in der Draufsicht dargestelltes Schlachtfeld mit diversen Gimmiks, wie etwa Geschützstellungen, gewürzt. Von allen Seiten stürmen gegnerische Soldaten auf ihn ein, denen es den Garaus zu machen gilt.
Die Qualität von Sound und Grafik hält sich in Grenzen, aber der Daumen tut Ihnen bestimmt schon beim ersten Außenposten vom vielen Feuern weh.

N.O.M.A.D.: Das eigenwilligste Ballerspiel in dieser Reihe ist wohl zweifelsohne N.O.M.A.D. Der Programmierer, auf dessen Konto dieses Spiel verbucht werden muß, ist zweifelsohne ein Freund von Abenteuer-Spielen. Denn eine nahe Verwandtschaft zu den Arcaden-Adventures kann N.O.M A.D. nicht leugnen. Es gilt, einen Kampfroboter durch die labyrinthartig verwobenen Gänge eines ausgehöhlten Asteroiden zu geleiten. In die Wände dieser interstellaren Festung, die übrigens in der Draufsicht dargestellt wird, sind an allen möglichen Ecken und Enden die raffiniertesten Abwehrwaffen eingebaut. So manche automatische Kanone, die in gewissen Intervallen durch die Gegend ballert, verriegelte Türen und gegnerische Kampfroboter machen das Vorankommen nicht gerade einfach. Auch ist die Steuerung des N.O.M.A.D.-Roboters ein wenig gewöhnungsbedürftig und leider etwas unsauber programmiert. Es kann bisweilen vorkommen, daß der Robot trotz vollem Rückwärtsschub nicht von einer Wand, an der er hängt, wegkommt. Ansonsten wäre noch eine feine Farbgrafik und das fast vollständiges Fehlen von Soundeffekten erwähnenswert.



N O.M.A.D.

Die Geschicklichkeitsspiele

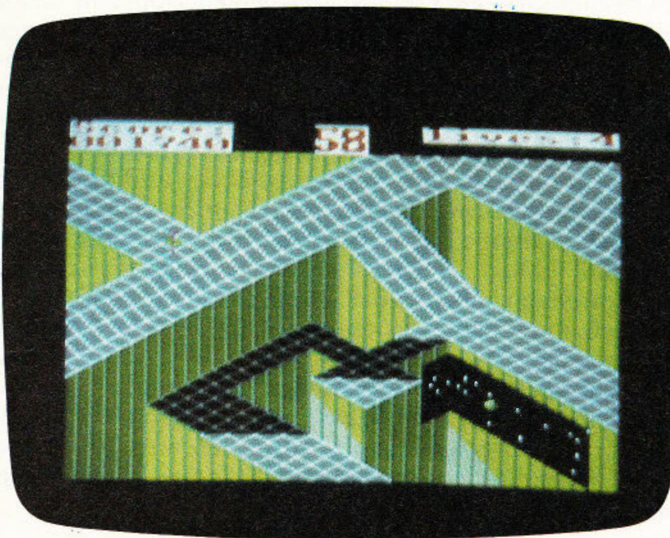


Thing on a Spring

THING ON A SPRING: Auf den ersten Blick vermeint man beim Betrachten von THING hier eher einen Vertreter der Jump & Run-Spiele vor sich zu haben. Was mich dazu bewog THING hier in die Kategorie der Geschicklichkeitsspiele aufzunehmen, war die unorthodoxe Fortbewegungsart der kleinen Spielfigur. Diese besteht nämlich, wie der Titel schon andeutet ,aus einem Kopf, zwei Armen und anstelle des Torsos und der Beine einer großen, quitschigen alten Bettfeder. Dieses kleine Federmännchen muß nun, vom Spieler gesteuert, ein Puzzle zusammensetzen.

Allerdings sind diese Teile in der Fabrik eines bösen Koboldes versteckt und eine ganze Menge seiner Spießgesellen bewachen die Teile. Die Fabrik, durch die der Federmann geistern muß, teilt sich in vier Etagen auf, die als Querschnitt dargestellt, durch einen Aufzug untereinander verbunden sind. Im Laufe der Suche findet man allerdings nicht nur Puzzleteile ,sondern auch andere Gegenstände, die der Spielfigur durch Aufnehmen zu einigen zusätzlichen Fähigkeiten verhelfen. Diese sind für ein gutes Gelingen der Suche aber unbedingt vonnöten.

Von der Grafik her ist THING sauber gemacht, alles ist

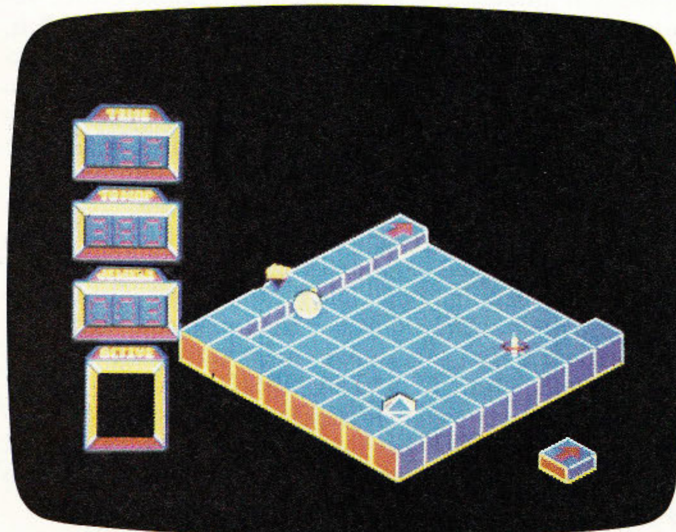


Gyroscope

bunt und phantasievoll in Szene gesetzt. In Verbindung mit dem bei diesem Spiel wirklich ausgezeichneten Sound ergibt das ein gelungenes Spiel, auf das man immer wieder gerne zurückkommt.

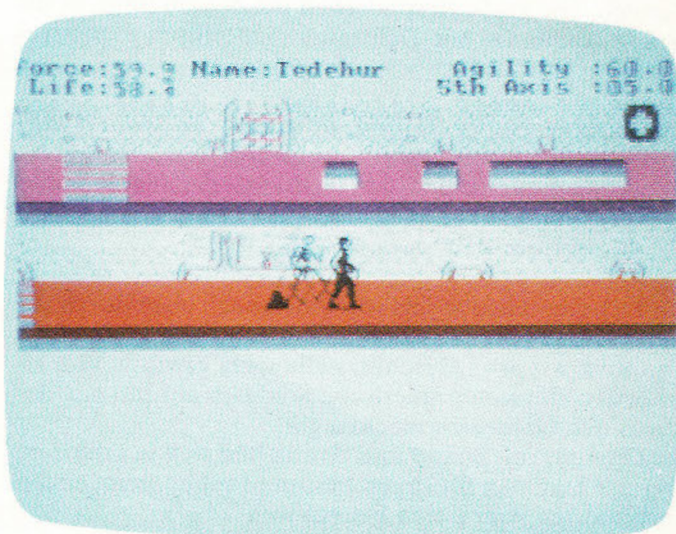
GYROSCOPE: Seitdem das wahnsinnige Kugelspiel "Marble Madness" die Spielhallen in Atem hält, grübeln die Softwarehersteller, wie sie ein derartiges Spiel auf diversen Heimcomputern realisieren könnten. Das erste Softwarehaus, dem eine solche, wenn auch nicht autorisierte, Umsetzung gelang, war Melbourne House. Um sogleich eventuell aufkeimendes Plagiatsgemunkel zu vermeiden, verzichtete man hier darauf, eine Kugel durch den Parkour zu jagen, und verlagerte sich auf einen Kreisel.

Aber ansonsten sieht GYROSCOPE eigentlich wie der kleine Bruder von Marble Madness aus. Das Ziel dieses Spieles ist es, einen kleinen Kreisel über einen verzwickten Parkour zu steuern, der stellenweise noch mit schiefen Ebenen oder etwa Gravofeldern verschärft wurde. Zwar ist GYROSCOPE beileibe kein schlechtes Spiel – hat es doch eine gute Grafik und mehrere verschiedene Musikstückchen, die während des Spiels ablaufen – der Mangel ist einfach der Spielumfang, und der ist mit nicht einmal 30 verschiedenen Bildern ein wenig dürfig, betrachtet man im Vergleich dazu die zweite Adaption von Marble Madness, die es für die CPC's gibt.



Spindizzy

SPINDIZZY: Spindizzy hält die Welt in Atem. Einigen Wissenschaftlern ist es gelungen, sich Zugang zu einer anderen Dimension zu verschaffen. Da es nicht möglich ist, einen Menschen in diese Zone zu versetzen, es dort allerdings einige besonders wertvolle Juwelen gibt, wurde ein Weg gefunden, in diese Dimension einzudringen und dort die begehrten Juwelen einzusammeln. Die speziell dafür entwickelte Spindizzy-Sonde wird daraufhin auf die Reise geschickt, um die Zone zu erkunden. Den Wissenschaftlern, die die Sonde aus sicherer Entfernung via Fernbedienung steuern, präsentiert sich eine sonderbare Welt, die aus schiefen Ebenen und schmalen waagerechten Stegen besteht. Aufgrund einer physikalischen Gesetzmäßigkeit, kann die Sonde nur eine bestimmte Zeit in der Zone verbleiben, danach gilt es, so schnell wie möglich zu verschwinden. Im Grunde genommen ist, abgesehen von der hier fehlenden Musikuntermalung, kein großer Unterschied zu GYROSCOPE. Allerdings hat SPINDIZZY über 250 verschiedene Bilder, die zusammen mit einigen Optionen die das Spiel vereinfachen, ein echtes Kleinod ergeben, das in keiner Softwaresammlung fehlen sollte.



The 5th Axis

THE 5TH AXIS: Wieder einmal hat sich ein Wissenschaftler mit der Entwicklung eines ganz besonders gewieften Gerätes hervorgetan, es ist ihm gelungen, eine Zeitmaschine zu bauen, mit der dann auch sogleich experimentiert wird. Allerdings stellt sich heraus, daß dieses Ding bei weitem noch nicht so arbeitet wie man sich das vorgestellt hat.

Zwar ist es mit der Maschine möglich, die Zeit zu manipulieren, allerding in einem Umfang, den sich niemand zu erträumen wagte. Die Zeitmaschine wirft alle möglichen verschiedenen Zeitzonen durcheinander und ein absolutes Chaos macht sich in den Labors breit. Das Spielgeschehen von THE 5TH AXIS wird auf dem Monitor als ein Querschnitt durch das mehrstöckige Laboratorium dargestellt. Der Spieler muß nun die unterschiedlichsten herumliegenden Gegenstände einsammeln und in ihre angestammte Zeitzone zurückbringen. Dabei muß man darauf achten, daß die amoklaufenden Hilfsroboter dem Spielfigürchen nicht das Fell über die Ohren ziehen. Auch erschweren Löcher in den einzelnen Stockwerken das Vorankommen, zwar ist es möglich, von tieferliegenden Eta-

gen mittels Lifts wieder nach oben zu fahren, aber mancherorts ist das Überspringen klaffender Abgründe unumgänglich.
Die Grafik, die THE 5TH AXIS sein Eigen nennt, ist zwar sehr detailliert, aber farblich nicht besonders attraktiv geworden. Trotzdem gestaltet sich das Spiel abwechslungsreich und unterhaltsam.

Sportspiele

3D-BOXING: Sich zu prügeln, aber gleichzeitig keine schmerzhaften Schläge einstecken zu müssen, war wohl schon immer ein Menschheitstraum. Wahr geworden ist er allerdings erst mit der Entwicklung von leistungsfähigen Homecomputern.
Wie der Name schon unmißverständlich verrät, geht es hierbei um's Boxen. Gegen die unterschiedlichsten Gegner gilt es im Ring eine gute Figur zu machen, und sie, wenn möglich, lang hinzustrecken. Je weiter der Spieler in der Rangliste vorankommt, desto härter wird der Fight im Ring. Denn selbstverständlich wachsen auch die Fähigkeiten Ihrer Widersacher.



3D-Boxing

Mit einer netten Grafik versehen und einer ganz passablen Sounduntermalung, ist dieses Spiel ein gut gelungener Vertreter der Sportsimulationen. Zwar hätte man mit ein wenig mehr Unterstützung durch einen guten Grafiker wesentlich mehr aus dem Spiel machen können, aber was nicht ist, kann ja noch werden.

WINTERGAMES: Dieses seit langer Zeit angekündigte Programm von Epyx erreichte den Markt erst mit einiger Verspätung. Da das Programm in seiner Commodore-Version ein absolutes "Non plus Ultra" für Grafik und Spielwitz darstellt, waren die Erwartungen, die der Schneider-User in die Umsetzung für die CPC's setzte, entsprechend hoch. Um diese hochgeschraubten Erwartungen dann auch befriedigen zu können, mußte ein enormer Programmieraufwand getrieben werden, und das kostete natürlich eine ganze Menge Zeit. Aber dafür kann sich das Programm, das nun schlußendlich dabei herausgekommen ist, auch wirklich sehen lassen.
Wie der Titel schon verrät, handelt es sich hier um die Simulation einer Winterolympiade, in deren Verlauf der Spieler versuchen muß, in verschiedenen Disziplinen



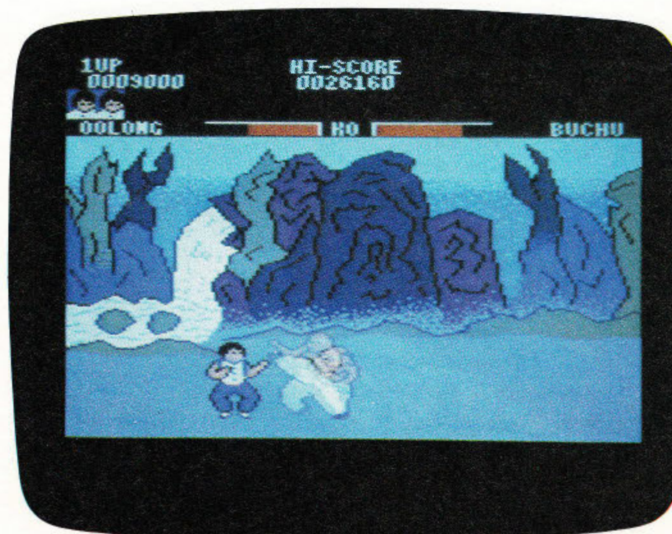
Wintergames

akzeptable Ergebnisse zu erzielen. Die Bandbreite der hier angebotenen Sportarten reicht vom Skispringen bis zum Biatlon. Im Gegensatz zu anderen Sportsimulationen, wurde hier großer Wert auf eine akkurate Steuerung gelegt. Der Spieler muß nicht wie ein verrückter am Joystick herumrödeln, sondern ihn rhythmisch im Takt zu den Bewegungen des Spielmännchens betätigen.
Grafik und Sound haben hier bisher nicht gesehene Qualitäten und machen aus Wintergames eine der besten Sportsimulationen, die es für die CPC's gibt.

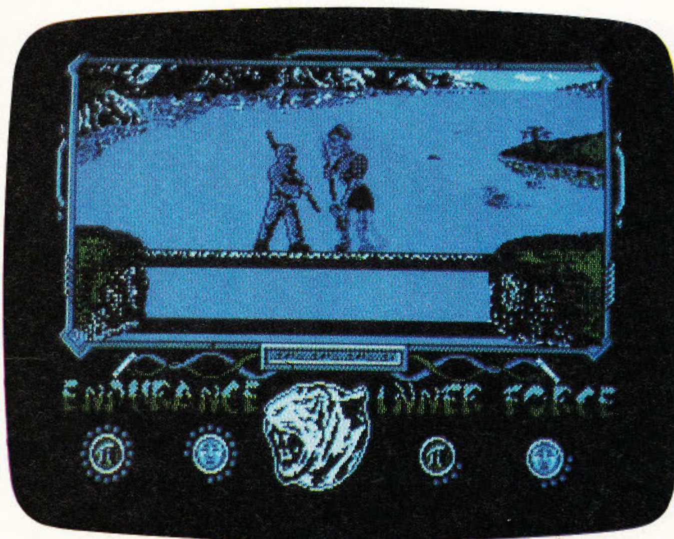
MARTIAL ARTS

YIE AR KUNG FU: Die Spiele des Martial Arts-Genre sind im Grunde genommen nahe Verwandte der Sportsimulationen, da es auch hier darum geht, eine Spielfigur, die wie ein Mensch aussieht, so zu steuern, daß sie jeweils die gewünschte Bewegung ausführt. Daß es hier allerdings nicht so sportlich zugeht wie bei den Sportspielen, versteht sich von selbst.

Sieht man sich YIE AR KUNG FU einmal genauer an, so vermeint man sich in einer computerisierten Version eines



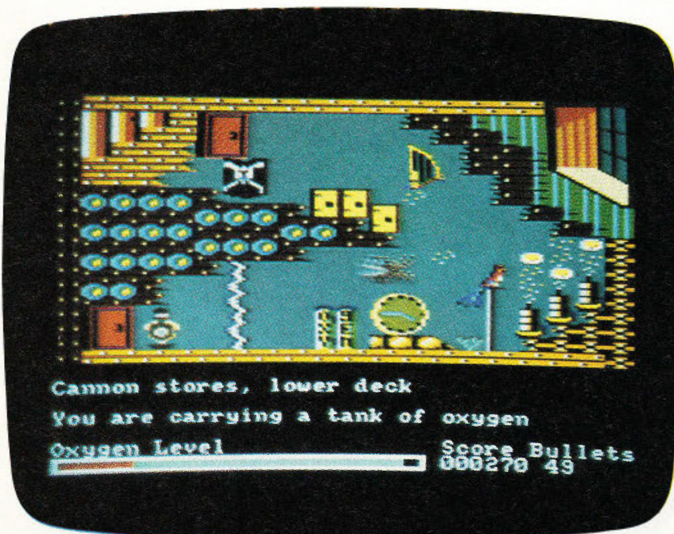
Yie ar Kung Fu



The Way of the Tiger

Karate Films zu befinden .Ziel des Spieles ist es ,den Gegner mit den hinterlistigsten Tricks aufs Kreuz zu legen. Diese sind aber leider meistens bewaffnet und wollen sich das gar nicht gefallen lassen. Um dem Geger ein blaues Auge zu verpassen ,stehen dem Spieler acht verschiedene Aktionen zur Verfügung, das können die Gegner allerdings auch und versuchen sich dementsprechend zu wehren. Mit meterhohen Sprüngen und gemeinen Tritten gegen jedes sich bietende Ziel wird da um den Sieg gekämpft. Die doch recht lustig gemachte Spielhandlung trägt ungemein zum Gefallen des Spieles bei, das auch nach mehrmaligem Einladen immer wieder Spa ß macht.

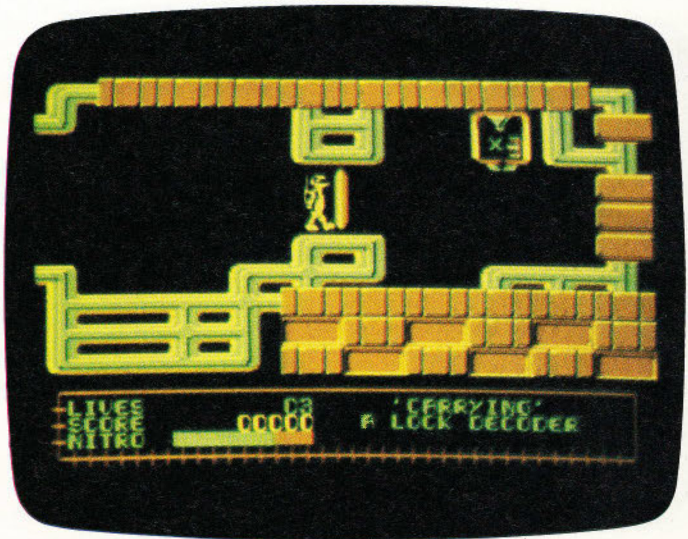
THE WAY OF THE TIGER: Auch hier geht es um Karate und die verwandten fernöstlichen Kampfkünste. Im Gegensatz zu YIE AR nimmt sich THE WAY allerdings um einiges ernster. In Disziplinen wie Stabfechten, Schwertkampf oder der einfachen Selbstverteidigung ohne Waffen steht man hier Gegnern gegenüber, die weniger zum Lachen denn zum Fürchten sind. Gräuliche Kobolde mit langen Messern oder wilde, fast unverwundbare Nashornmenschen stellen sich einem weiteren Vordringen in den Weg.
Von der Grafik her ist THE WAY sicherlich das am besten gelungene Martial Arts-Spiel, das es für den Schneider überhaupt gibt. Das Spiel findet in einem dreifach gescrollten Szenario statt. Das heißt, daß sich der Vordergrund, die Mitte des Bildes und der Hintergrund unabhängig voneinander bewegen. Daß dem Programmierer eher eine Mischung aus Kung Fu und Fantasy vorschwebte, wird sehr schnell klar. Soundmäßig ist auch einiges zu dem Programm zu bemerken: ein kleines, richtig fernöstlich klingendes Musikstück ist in das Spiel mit eingebaut worden, das allerdings nur ertönt, während das Menue auf dem Monitor zu sehen ist. Während des eigentlichen Spieles sind dann nur die obligatorischen Kampfgeräusche zu hören. Trotzdem ist THE WAY sicherlich eines der besten Karatespiele überhaupt.



The Devils Crown

Arcaden Abenteuer

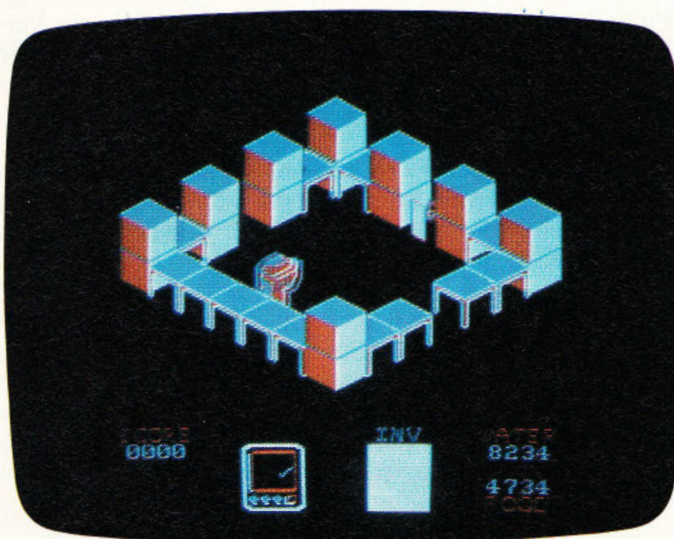
THE DEVILS CROWN: Nicht nur die Sektion der einfachen Textabenteuer befindet sich derzeit rapide auf dem Vormarsch. Auch der kleine Bruder, die Arcaden-Adventures erfreuen sich immer größerer Beliebtheit beim Publikum. Einer der farbigsten Vertreter dieses Genres ist DEVILS CROWN. In diesem Spiel hat der Spieler die Aufgabe das Wrack eines gesunkenen Schiffes zu erkunden.



Obsidian

Im Inneren dieses Wracks treiben sich allerdings immernoch, wohl aufgrund diverser Flüche, die toten Piraten herum, und versuchen, dem Spieler, der das Wrack zu erkunden sucht, das Leben schwer zu machen. Das Innere des alten Schiffes besteht aus einer Vielzahl verschiedener Räume, die in einer durchschnittlich guten, aber manchmal recht interessant gemachten, Farbgrafik dargestellt werden. Manche der Türen sind verschlossen und können erst unter Zuhilfenahme bestimmter Gegenstände geöffnet werden. In anderen Räumen ist es stockdunkel und erst eine mitgebrachte Lampe macht das Passieren des Zimmers möglich. Fazit: Ein gut gemachtes Arcaden-Abenteuer, das zwar nicht zur Elite des Genres zählt, aber durch seine saubere und klare Machart zu überzeugen versteht.

OBSIDIAN: Wie schon bei anderen Spielen vorher, wurde für OBSIDIAN ein ausgehöhlter Asteroid gewählt. Im Inneren dieses kosmischen Gesteinsbrockens ist ein ganzes wissenschaftliches Institut errichtet worden, dessen Aufgabe es sein sollte, ein Schwarzes Loch zu studieren. Ein Defekt an den Antriebsaggregaten droht das ganze



Chimera

Projekt scheitern zu lassen. Ein einzelner Mann muß sich in den Asteroiden begeben und dort alle ausgeschalteten Anlagen wieder in Betrieb setzen. Bedauerlicherweise hat der Held des Spieles vergessen, sich die einzelnen Aktionen, die zur Wieder-Inbetriebnahme notwendig sind, erläutern zu lassen. Durch die unmittelbare Nähe des Schwarzen Loches ist an eine Funkverbindung nicht zu denken, also muß der Held nun ganz auf sich alleine gestellt herausbekommen, wo was womit ein- oder ausgeschaltet wird.

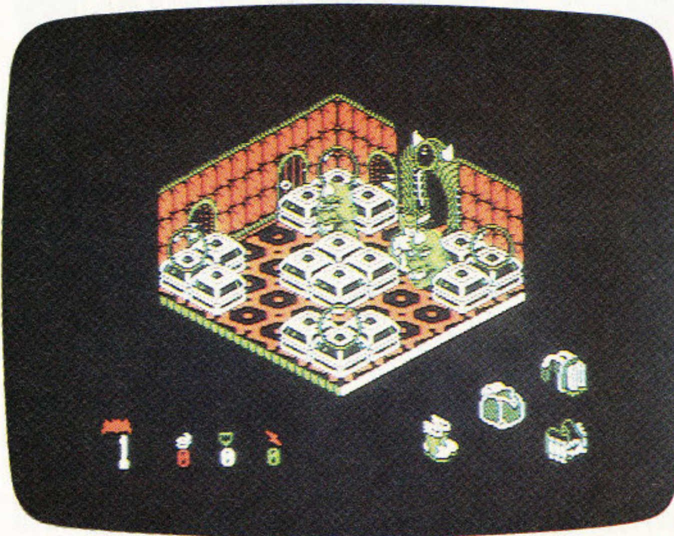
OBSIDIAN zeigt recht deutlich, welche qualitativ hochwertigen Grafiken auf den CPC's möglich sind. Vom Sound her liegt die Sache leider, wie so oft, im Argen. Denn das, was da aus dem Lautsprecher des CPC's quillt, ist fast gar nichts, man erschrickt regelrecht, wenn wirklich einmal ein Geräusch ertönt. Aber ansonsten kann das Spiel weiterempfohlen werden.

CHIMERA: Das Ultimate nicht das einzige Softwarehaus ist, das über das notwendige Know How verfügt, Spiele mit 3D-Touch zu programmiern, wurde spätestens mit CHIMERA klar. Erstaunlich ist auch, daß ein so gut gelungenes Spiel unter einem Budget Label vertrieben wird. Die Spielhandlung dreht sich um ein extraterrertrisches Raumschiff, das plötzlich von irgendwoher in unser heimatliches Sonnensystem eindringt. Untersuchungen ergeben, daß das fremde Schiff auf die Erde zu stürzen droht. Es muß also etwas unternommen werden.

Ein einzelner Mann dringt in das Innere des Geisterschiffes ein und versucht, die schiffseigene Selbstzerstörungsanlage in Betrieb zu setzen. Wie bereits erwähnt, erscheint CHIMERA auf dem Screen in einem gefälligen 3D-Outfit.

Das wesentliche Spielmoment ist hier das Einsammeln und Kombinieren herumliegender Gegenstände. Wie so oft, ist auch hier der Sound so schlecht wie die Grafik gut ist. Zu bemerken wäre allerdings noch, daß CHIMERA nicht gerade eines der leichtesten Spiele seiner Art ist.

BATMAN: Da wir schon einmal bei den Spielen, die sich im 3D-Gewand präsentieren, sind, machen wir sogleich mit einem ähnlichen Spiel weiter.

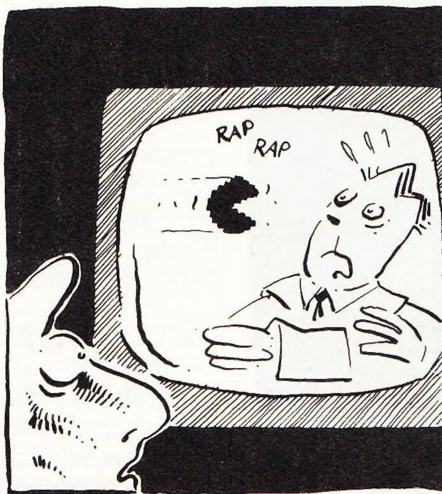


Batman

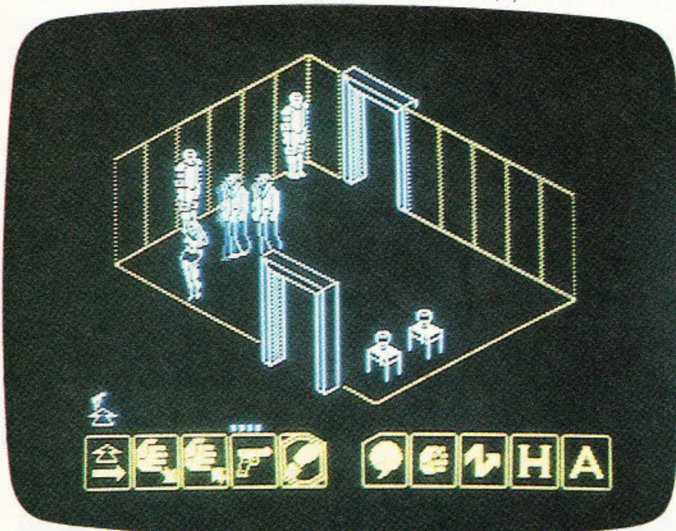
Zwar wird bei BATMAN ein wenig mehr Wert auf das Arcaden-Moment gelegt, aber trotzdem erfüllt es alle Ansprüche eines Abenteuers. BATMAN ist schon seit Jahrzehnten einer der beliebtesten amerikanischen Superhelden. Sein erstes Betätigungsfeld waren die Comics. Durch den großen Erfolg, den er dort hatte, wurden auch die englischen Softwareproduzenten auf ihn aufmerksam. Danach war es nur noch eine Frage der Zeit, bis ein Computerspiel mit ihm als Hauptfigur realisiert wurde.

Zwei seiner ältesten Widersacher sind ohne sein Wissen in die Bathöhle eingedrungen und haben seinen Side-Kick







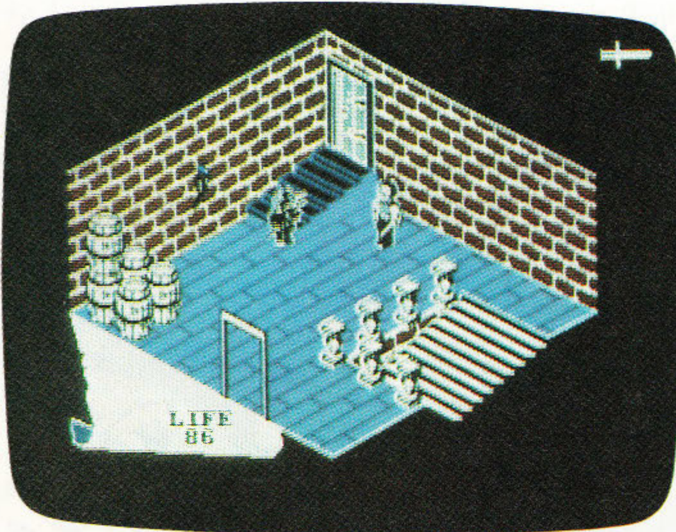


Movie

Robin, den Wunderboy entführt. Damit Batman ihnen nicht folgen kann, haben sie sein Auto, das Batmobil, in seine Einzelteile zerlegt und es überall in der Bathöhle versteckt. Um Robin zu retten, muß Batman zuerst einmal sein Auto wieder zusammensetzen. Mit über 150 verschiedenen Räumen ist BATMAN ein Spiel mit einer recht großen Spielfläche. Vom Schwierigkeitsgrad her läßt sich sagen, daß es zu Beginn des Spieles recht einfach ist, aber je tiefer man in die Bathöhle eindringt, desto schwieriger wird es. Alles in allem ein spannendes und gut gemachtes Spiel.

MOVIE: Qualitativ mindestens ebensogut wie Batman, ist MOVIE. Ein Spiel, das auch 3D-Grafiken verwendet, allerdings in einer recht eigenwilligen Art und Weise. Denn hier wird weniger mit Vollflächen gearbeitet, sondern mit eher vectorartigen Linien, durch die die Außenwände der Räume dargestellt werden.

Die Objekte, die sich in den einzelnen Räumen befinden, sind mit viel Liebe zum Detail erstellt worden. Überhaupt ist die Grafik von MOVIE absolut gelungen; stellenweise glaubt man weniger an einem Computerspiel zu sitzen, sondern eher sich einen Comicstrip anzusehen. Die Story, um die es hierbei geht, ist ein hochkarätiger Krimi, in dem ein Privatdetektiv den Auftrag erhält, die Zwillingsschwester eines Gangsterliebchens zu suchen. Allerdings ist der



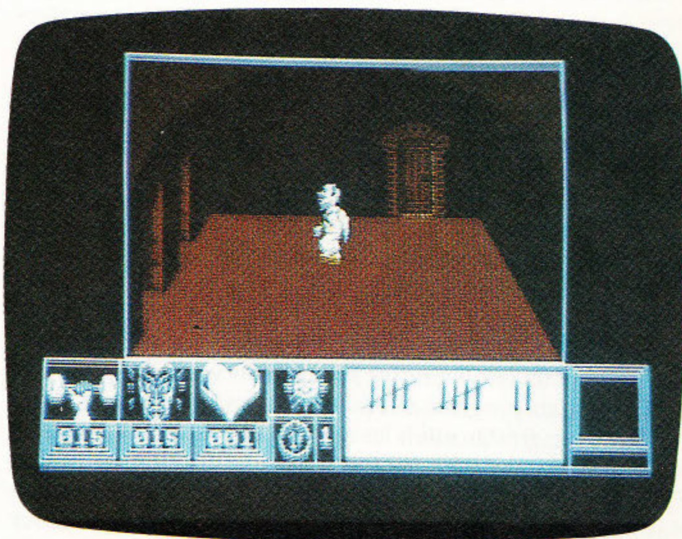
Fairlight

Freund der Gangsterbraut einer der gefährlichsten Verbrecher der Stadt. So hat denn auch unser Privater alle Hände voll zu tun, heil aus der ganzen Sache herauszukommen.

FAIRLIGHT: Über die tollen 3D-Grafiken dieses Spieles erzähle ich Ihnen am besten erst gar nichts. Sehen Sie sich das Bildschirmfoto dazu an, und Sie wissen sofort, was Sie erwartet.

Abgesehen von seiner edlen Grafik ist das sehr ausgeprägte Abenteuermoment dieses Spieles erwähnenswert. Durch eine gewaltige Auseinandersetzung der Mächte von Ordnung und Chaos ist das Licht der Sonne erloschen. Das Land von Fairlight lebt seither im andauernden Zwielicht. Allerdings lebt ein Magier, der bei diesem Kampf eine wesentliche Rolle spielte, immer noch.

Zwar liegt sein Körper in tiefen Schlaf versetzt in einem der Keller der Burg Avar, doch sein Geist ist wach und er kann ihn umherwandern lassen. So geleitet er einen jungen Mann in die Burg, die seit tausend Jahren kein Mensch mehr betreten hat. Dort erzählt er ihm, daß der einzige Weg, die Burg wieder wohlbehalten zu verlassen, darin besteht, ein verborgenes Zauberbuch zu finden. Mit diesem Zauberbuch wäre es dem Magier dann auch möglich, das Zwielicht aus dem Lande FAIRLIGHT zu verbannen.



Eden /Doomsday Blues

Dieses Spiel ist mit Sicherheit eines der besten Arcaden-Abenteuer, die es für den CPC gibt. Joyce-Besitzer wird sicherlich interessieren, daß es FAIRLIGHT auch in einer Joyce-Version gibt, die, von der Farbe einmal abgesehen, der CPC-Version um nichts nachsteht.

EDEN / DOOMSDAY BLUES: Einen recht eigenen 3D-Touch haben die Programmierer diesem Spiel mit auf den Weg gegeben. Dieses mal wurde eine Perspektive mit Fluchtpunkt in der Bildschirmmitte gewählt. Das Szenario verjüngt sich perspektivisch nach hinten.

Die Handlung findet im Inneren eines menschenleeren Gefängnisses statt. Nachdem die Roboter alles Leben auf der Welt vernichtet haben, halten sie sich einen letzten Menschen als Schaustück. Irgendwann einmal hört dieser Mann das Schluchzen einer Frau. Als er sich in einem verzweifelten Wutanfall gegen die Tür wirft, stellt er fest, daß

diese nicht verschlossen ist. Aber was soll er nun mit seiner wiedergewonnenen Freiheit anfangen? - Richtig. Er macht sich auf die Suche nach der Frau, deren Schluchzen er gehört hat.

EDEN BLUES ist eines der ersten französischen Spiele, die zu uns nach Deutschland gekommen sind. Die neuartige Präsentation auf dem Bildschirm gibt diesem Spiel einen eigenen Reiz.

Zusätzlich gewinnt das Spiel durch einen kleinen Kunstgriff an Attraktivität. Die Charakteristiken der Spielfigur werden durch Zahlwerte dargestellt, sinkt einer dieser Werte auf Null, bedeutet das den Tod des Männchens. Während des Spieles hat der Spieler allerdings immer wieder die Möglichkeit, seine Werte mittels gefundener Gegenstände, wie etwa Weinflaschen, aufzustocken.
Auf jeden Fall gehört EDEN BLUES zur Oberliga der CPC-Spiele.

SORCERY: Ein Spiel, über das man selbst bei nüchterner Betrachtung ins Schwärmen kommen kann. Was hier auf dem Monitor stattfindet, ist allererste Obersahne. Die Spielstory rankt sich um einen Zauberer, der seine von einem böswilligen Konkurrenten gefangengehaltenen Schüler befreien muß. Dazu muß er allerdings zuerst einmal den jeweils richtigen Gegenstand einsammeln.

Ein Zeitlimit, das durch ein langsam verschwindendes Buch symbolisiert wird, macht aus dem Spiel ein Rennen gegen die Zeit. In Burgen, Schlössern, unterirdischen Verliesen und alten Ruinen, die allesamt in einer wahnsinns Farbgrafik gemacht wurden, sind die benötigten Gegenstände versteckt und werden zu allem Überfluß auch noch von ganzen Dämonenhorden bewacht.

Nachdem sich SORCERY in seiner Kassettenversion als ein absoluter Hit entpuppt hat, ist das Programmiererteam, das auf den interessanten Namen "Gang of Five" hört, darangegangen, eine spezielle Disk-Version von dem Spiel zu erstellen. Aus dieser Disk-Version ist aber dann mehr geworden.

Abgesehen davon, daß in den eigentlichen Spielverlauf noch einige Extra-Räume integriert wurden, haben die Programmierer einen zusätzlichen Bereich eingebaut, in den man gelangt, nachdem alle Zauberlehrlinge befreit wurden. In dieser zusätzlichen Sektion kommt es dann zur



Sorcery



Strangeloop

großen Auseinandersetzung zwischen den Mächten des Guten und den Kräften der Finsternis.

SORCERY ist ein absoluter Klassiker für die CPC's, obwohl es nun schon fast ein Jahr auf dem Buckel hat, sollte sich jeder Spieler, der es noch nicht sein Eigen nennen kann, unbedingt zulegen. Um sich dann auch gleich von Anfang an in dem Labyrinth zurechtzufinden, sollten Sie auch die Seiten 41 bis 44 beachten: dort finden Sie eine detailreiche Karte von Sorcery.

STRANGELOOP: Daß die Gang of Five nach ihrem SORCERY-Erfolg nicht untätig geblieben ist, belegt das Spiel STRANGELOOP.

Wieder einmal geht es um einen ausgehöhlten Asteroiden, in den eine automatische Fabrik hineingebaut wurde. Diese Fabrik produziert Arbeits- und Haushaltsroboter. Als plötzlich die große Zentralrechenanlage ausfällt, oder zumindest ein bißchen verrückt spielt, laufen nur noch Robbies mit falscher Programmierung vom Band. Das neue Programm macht aus den kleinen Helferlein eine Horde von mordlustigen Maschinen, die fortan überall auf der Welt ihr Unwesen treiben. Ein Zustand, der behoben werden muß.

Wieder einmal muß ein Selbstmordkandidat her. Er muß in die Fabrik eindringen und dort die zur Vernichtung des Zentralrechners notwendigen Gegenstände einsammeln. Zwar hat auch STRANGELOOP eine ganz passable Grafik, aber an die Qualitäten des Vorgängers kommt sie nicht heran; auch hat sich in punkto Spielgeschehen nicht besonders viel verändert.

Gut ist STRANGELOOP allemal, aber ist man schon der stolze Besitzer einer SORCERY-Kopie, kann man sich STRANGELOOP getrost sparen.

THREE WEEKS IN PARADISE: Dieses Spiel steht stellvertretend für die ganze Wally Week-Serie. Die Week Familie, bestehend aus Wally, Wilma und little Herbert, macht Urlaub in der Südsee. Aber schon am ersten Tag passiert ein großes Malheur. Das Boot, das sie zu ihrem Hotel bringen sollte, setzt sie irrtümlicherweise auf einer anderen, immer noch von Kannibalen bewohnten Insel ab. Ruckzuck sind Herbert und Wilma in der Gewalt der Eingeborenen, die aus ihnen noch am selben Abend ein



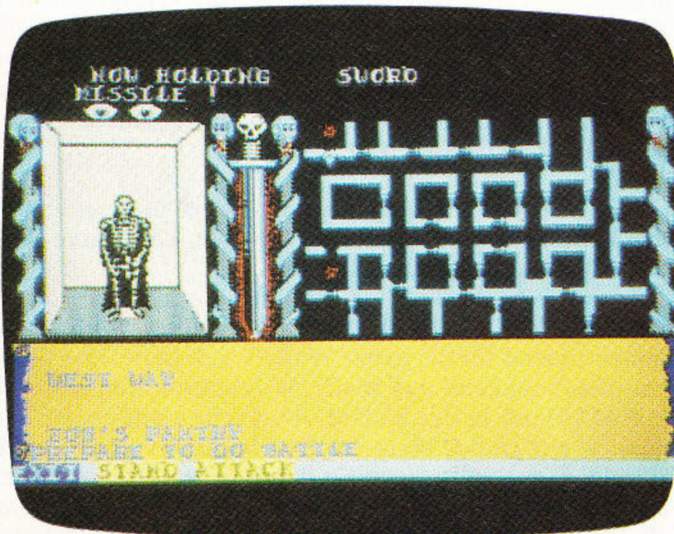
Three Weeks in Paradise

großes Festmahl machen wollen. Der arme Wally hat nun die Aufgabe, seine Lieben zu befreien, aber bis dahin ist noch ein langer Weg.

Micro Gen, die Firma, die THREE WEEKS realisiert hat, scheint an der Week Familie einen Narren gefressen zu haben. Immerhin gibt es schon fünf verschiedene Spiele um diese Chaostruppe.

Automania, Pyjamarama, Everyones a Wally, Herberts Dummy Run, und eben Three Weeks in Paradise.
Von diesen fünf Spielen sind alle, abgesehen von Automania, auch für die CPC's zu haben. Diese vier gleichen sich vom Spielgeschehen wie ein Ei dem anderen. Jedesmal gilt es ein, zwar immer anderes Endziel zu erreichen, aber der Weg, wie man an dieses Ziel kommt, unterscheidet sich durch nichts.
Wiederholt macht man sich auf die Suche nach Gegenständen, durch die man, eingesetzt an den richtigen Stellen, ein Stückchen weiter kommt.

Fazit: Grafik bunt und gefällig, Sound nicht oder nur kaum auffindbar, Spielwitz: hält sich in Grenzen. Wen es interessiert, der sollte sich eines zulegen, und er wird sicherlich eine ganze Menge Spaß damit haben, aber sich dann auch noch ein weiteres zuzulegen, ist unnötig.

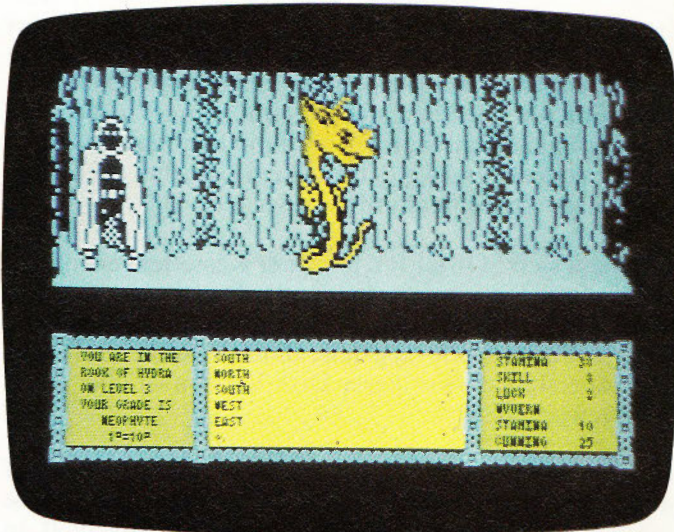


Swords an Sorcery

SWORDS AND SORCERY: Ein Arcaden-Adventure ganz anderer Machart stellt sich mit diesem Spiel vor. Von der Struktur her sehr eigenwillig aufgemacht, ist SWORDS AND SORCERY eines der besten Spiele, die es in 1986 gab. Interessant ist die Art und Weise, wie das Spiel gesteuert wird. Alle wesentlichen Aktionen werden über Menues angewählt und abgerufen.

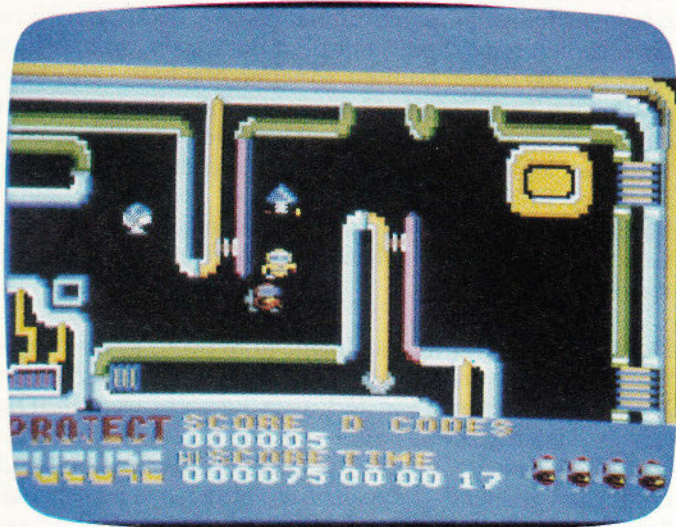
Die Geschichte, die diesem Spiel als Hintergrund dient, ist die abenteuerliche Suche nach einem Ausweg aus einer alten, labyrinthartig verwinkelten Grabstatt. Durch diese alten Katakomben führt der Weg des Helden an allen nur vorstellbaren Ungeheuern, Monstren und sonstigen unheimlichen Wesen vorbei.

Die grafische Präsentation des Ganzen hat in dieser Form kein Beispiel. Der Monitor ist in verschiedene Windows eingeteilt, in denen alle für den Spieler zugänglichen Informationen abzulesen sind. In einem der Windows wird dann auch der jeweilige Ausblick, der sich dem Spieler gerade bietet, angezeigt.
Die Monster und Widersacher werden dabei in einer gelungenen Grafik teilweise animiert dargestellt. Sound ist auch hier, wie so oft, ein absoluter Mangel. Aber obwohl es schade ist, daß sich diesbezüglich nichts regt, tut das dem Spielen keinen Abbruch. Prädikat: sehr unterhaltsam und gut gemacht.



Heavy on the Magick

HEAVY ON THE MAGICK: Dieses Spiel glänzt durch ein, auf seine Art und Weise, gelungenes Konzept. Aufgrund einer fehlerhaft gesprochenen Beschwörung, wird Axil, der Fähige, in die Gänge einer alten Burgruine versetzt. Dort wimmelt es, wie sollte es anders sein, von schier abscheulichen Wesen, denen dann auch nichts Besseres einfällt, als unserem kleinen Zauberer nachzustellen. Erst nachdem Axil ein altes Zauberbuch gefunden hat, ist er in der Lage, sich des unheimlichen Getiers zu erwehren. Eine Besonderheit dieses Spieles ist, daß nicht alle der herumlaufenden Monster von sich aus bösartig sind. Mit einigen kann man sogar eine Unterhaltung führen, bei der so manche interessante Information zu Tage kommt. Einer der gesprächigsten Katakombenbewohner ist der baumlange Oger Apex. Er ist eine wahre Quelle brauchbarer Informationen. Die Grafik ist zwar sehr grob, aber dafür ausgezeichnet animiert. Zum guten Schluß möchte ich noch bemerken, daß HEAVY ON THE MAGICK eines der schönsten, aber auch eines der schwersten Abenteuerspiele ist, die ich seit langem gesehen habe.



Project Future

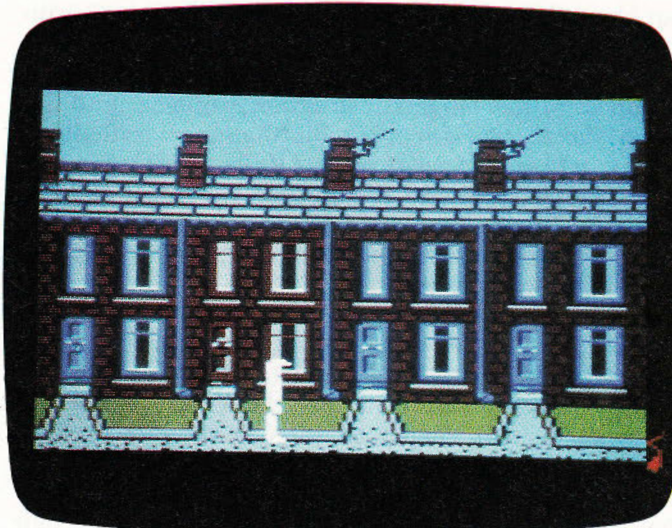
PROJEKT FUTURE: Die U.S.S. Future ist ein riesiges altes Cargoraumschiff. An Bord befinden sich tausende und abertausende Container mit Biomaterial. Als dieses Schiff nach langer interstellarer Reise zur Erde zurückkehrt, müssen die Kontrollstellen der irdischen Raumüberwachung feststellen, daß sich aus dem Biomaterial aufgrund der harten Strahlung, die im Weltraum vorherrscht, ganze Armeen von hirnlosen, aber dafür umso blutrünstigeren Mutanten entwickelt haben.

Zu ihrem Entsetzen erkennt die Flugleitstelle, daß die eigentliche Programmierung der Future eine Landung auf der Erde vorsieht. Wenn das geschieht, wäre das eine echte Katastrophe. Die einzige Möglichkeit, die Gefahr zu bannen, besteht darin, die Future zu vernichten. Ein Mann muß sich an Bord begeben und dort die automatische Selbstvernichtung des Schiffes einschalten.

Abgesehen von der mindestens schon hundert Male verwendeten Hintergrundstory bietet PROJEKT FUTURE auch sonst nicht viel Neues. Die manchmal unzulängliche Programmierung vergällt dem geneigten Spieler auch den letzten Spaß an diesem Programm. Das einzige, das Gremlin Graphics hier zu seiner Ehrenrettung vorbringen kann, ist der Umstand, daß dies eines ihrer ersten Spiele für den CPC war.
Ein Spiel also, das man sich lieber nicht zulegen sollte.

FRANKIE GOES TO HOLLYWOOD: Seit langer Zeit liegt hier endlich mal wieder ein Spiel vor, daß sich von seiner Machart her wohltuend von den üblichen Arcaden-Abenteuern abhebt. Nach der gleichnamigen Popgruppe benannt, bemüht der Hersteller sich auch gar nicht erst lange mit der Konstruktion einer Hintergrundstory. Es geht darum, einer Unperson dabei zu helfen, zu 100% eine richtige Person zu werden. Dies geschieht durch das Einsammeln und Anwenden bestimmter Gegenstände. Auch liegt in den Häusern, die es zu durchwandern gilt, ein Toter. Durch die Ermittlung des Täters kann der Menschwerdung des Spielfigürchens ein weiterer Schub gegeben werden.

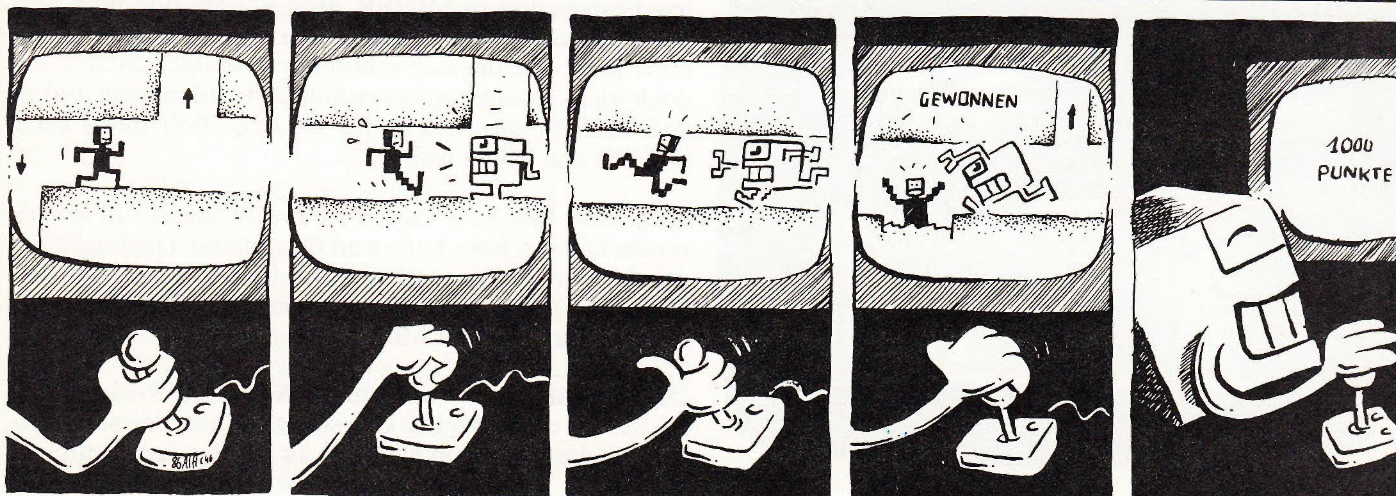
Hat der Spieler es dann geschafft, und die 100% erreicht, wartet eine weitere Aufgabe auf ihn. In einem unterirdischen Labyrinth muß der Pleasure Dome gefunden werden, um damit auch zum Ende des Spieles zu gelangen. Eine gute Grafik, eine noch bessere Spielhandlung ,feine Musikuntermalung und noch einiges mehr, lassen FRANKIE weit über den Durchschnitt der übliche Software-Qualität hinauswachsen. Auch die Tatsache, daß niemand versucht hat das Spiel auf irgendeine Art und Weise in ein am Ende doch nicht passendes Handlungsgerüst einzubauen, fällt hier wohltuend auf.

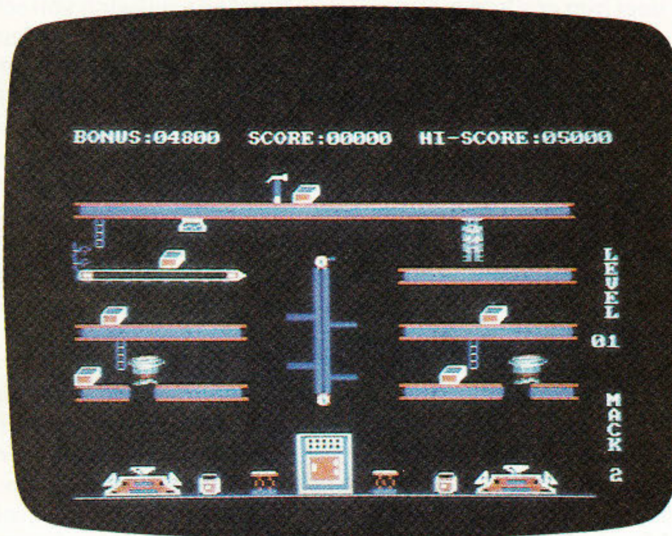


Frankie goes to Hollywood

Jump and Run

HARD HAT MAC: Ein echtes Wolkenkratzer-Schau-





Hard Hat Mac

stückchen hatte es sicher geben sollen; am Ende herausgekommen ist ein Spiel, das man sich nur mit einem lachenden und einem weinenden Auge ansehen kann.

Hart Hut Michael ist ein Bauarbeiter, der auf einer örtlichen Großbaustelle für die Erstellung der Stahlträgergerüste verantwortlich ist. Er muß Bolzen einsetzen, fehlende Teilstücke in die Trägerkonstruktion einfügen, und das Ganze dann mit einem Preßlufthammer festballern. Dabei macht er eine ganze Menge Radau, durch den sich einige Nachbarn gestört fühlen und außerdem ruft der Lärm eine Truppe von Vandalen auf den Plan, die fortan über das Gerüst geistern und versuchen, Micky von demselben herunterzuwerfen.

HARD HAT MACK war das erste Programm der renomierten Electronic Arts Programmierer, das von einem englischen Programmiererteam für die CPC's umgesetzt wurde. Zieht man Vergleiche zur Commodore-Version, stellt der Anwender umgehend fest, daß die Umsetzung gelungen ist. HARD HAT MACK ist auf dem CPC mindestens genauso schlecht wie das Original. Nein, nicht daß das Spiel durch und durch übel wäre, das Konzept ist im Grunde genommen ja auch nicht schlecht – aber das, was die Programmierer dann daraus gemacht haben, ist schlicht und ergreifend ein phantasie- und lieblos hingeklatschtes Spiel.

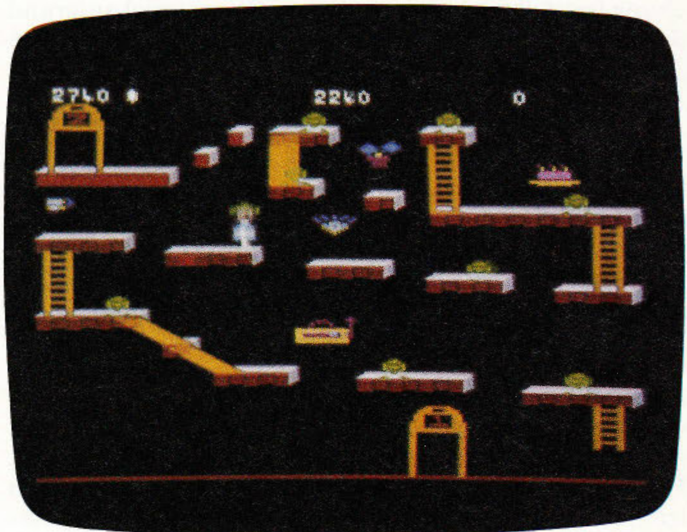


Bruce Lee

BRUCE LEE: Ah, das ist ein Spiel, über das ich gerne schreibe, denn, das gleich einmal vorweg, BRUCE LEE ist ein wirklich gutes Spiel, mit dem auch zwei Spieler ihre helle Freude haben können. Möglich macht das eine ausgeklügelte Zwei-Spieler-Option, auf die ich später noch einmal zurückkommen werde. Der Spieler schlüpft in die Rolle des leider verstorbenen Karate-Idols BRUCE LEE. In dieser Rolle hat er den Auftrag, in den Tempel einer bösen Sekte einzudringen und dort den finsteren Kult zu zerschlagen, indem er dem Oberbösewicht den Garaus macht.

Doch um dies zu bewerkstelligen, muß Bruce sich erst einmal durch die verschiedensten Räume kämpfen, die von Ninjas und anderen Finsterlingen bewacht werden. Die bereits erwähnte Zwei-Spieler-Option läßt sich so einstellen, daß der zweite Spieler, der normalerweise dasitzt und dem ersten Spieler dabei zusieht, wie er sich mit seinen Gegnern herumschlägt, eben die Rolle einer dieser Widersacher übernimmt.

Die Bewegungsmöglichkeiten entsprechen in etwa einem Karate-Spiel, aber das sehr ausgeprägte Laufen und Klettern hat BRUCE LEE dazu verholfen, hier in der Kategorie Jump and Run aufgenommen zu werden.



Bounty Bob strikes back

Ein Spiel, das auch bei längerem Spielen nichts von seinem Reiz verliert.

BOUNTY BOB STRIKES BACK: Dies ist die langerwartete Fortsetzung zu MINER 2049. Bounty Bob muß sich nach einem Atomkrieg durch eine alte, verlassene Uranmine kämpfen, die sich schon nach wenigen Metern als doch nicht so unbewohnt entpuppt. Hordenweise gibt es dort tödliche Mutanten, die unserem Bob dann auch sogleich ans Leben wollen.

Außerdem hält sich Yukon Johran in diesem Bergwerk versteckt. Er ist Bobs Leib- und Busenfeind. Die Jagd nach den begehrten Gegenständen und nach Yukon Johran führt Bob durch immer abenteuerlichere und gefährlichere Stockwerke der Mine.

Da es den Miner 2049 nicht für die Schneider Rechner gab, ist dies die erste Möglichkeit, sich zumindest diesen zweiten Teil des fast schon klassisch zu nennenden Plattformdramas zu genießen. Wie im allgemeinen bei derartigen

Spielen üblich, muß auch hier eine ganze Menge Geschicklichkeit bewiesen werden, um weiterzukommen. Um in den nächsten Level zu kommen, muß Bob erst alle Aufgaben erfüllen, die ihm in dem Level, in dem er sich gerade befindet, begegnen. Diese Aufgaben bestehen zumeist einfach darin, alle Stellen des Gerüstes einmal zu betreten.

BOUNTY BOB ist ein, von seiner Machart her klassisches Jump and Run Spiel, bei dem Geschicklichkeit und gutes Timing unumgänglich sind. Zwar kommt es hier meist nicht auf die Pixelgenauigkeit des Absprungs- bzw. Landepunktes an, um Bob springenderweise über das Gerüst zu lotsen, BOUNTY BOB deswegen gleich als ein leichtes Spiel zu bezeichnen, wäre falsch. Alles in allem ist das Spiel nicht schlecht gemacht, und der Freund guter Plattformspiele wird sicherlich seinen Spaß daran haben.

BOMB JACK: Nachdem der Halbsuperheld BOMB JACK fast jedem, der sich ab und an einmal in eine Spielhalle begibt, ein Begriff sein sollte, ist die Firma Elite Systems darangegangen, die explosiven Abenteuer BOMB JACK's auch für diverse Heimcomputer, unter anderem auch für den CPC, zu realisieren.

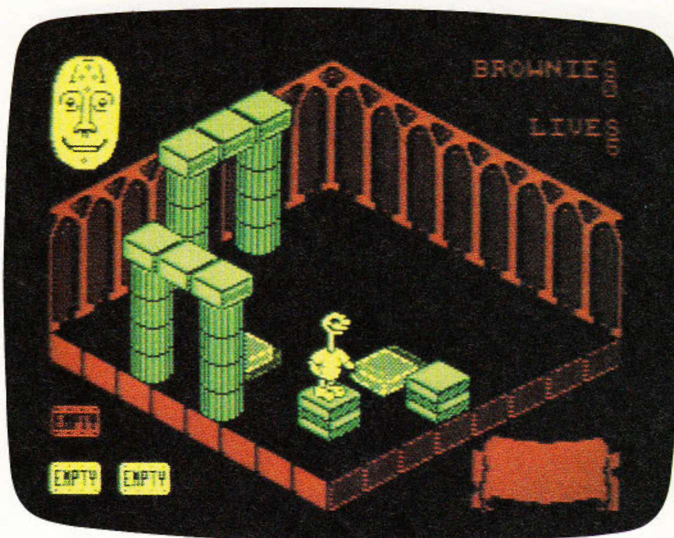


Bomb Jack

Primär gilt es, Bomben, die überall im Bild herumliegen, zu entschärfen. Doch dabei versuchen einige unfreundliche Zeitgenossen unseren lieben Bomben Jakob zu behindern, oder ihm, wenn es ihnen gelingt, durch Berühren einfach eines seiner drei Leben auszublasen.

Je weiter der Spieler die Scoreleiter hochklettert, desto schneller und aggressiver werden die kleinen Unholde. Teils hinderlich und teils von Nutzen, sind die verwinkelten Plattformen, über die Jack, der über eine enorme Sprungkraft verfügt, hüpfen muß, um an die brennenden Lunten der Bomben heranzukommen.

Wie bei den reinen Arcadenspielen üblich, geht es hier hauptsächlich um einen guten Score, den der Spieler aber erst mit einer gehörigen Portion Geschicklichkeit erreichen kann.
Die Steuerung ist ein kleines bißchen gewöhnungsbedürftig, macht aber schon nach kurzer Zeit keine Probleme mehr. Spielen kann man Bomb Jack ohne weiteres auch über längere Zeiträume, am meisten Spaß macht es, wenn



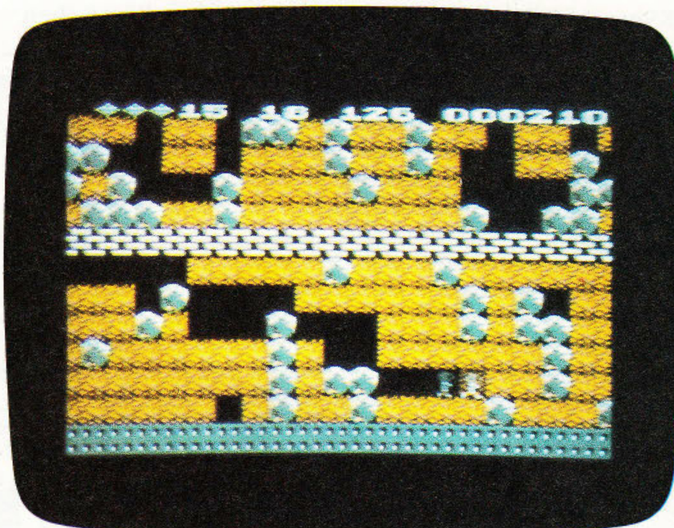
Sweevo's World

es mit mehreren Leuten gleichzeitig gespielt wird – eine solche Highscorejagd kann sich sehr unterhaltsam gestalten.

SWEEVO'S WORLD: Auf einem kleinen Planeten am Rande der Milchstraße existiert eine Rasse von kleinen Robotern, die durch die Bank hochintelligent sind. Nur der kleine Sweevo ist völlig aus der Art geschlagen, er ist nämlich strohdumm. Nachdem er durch so ziemlich jede Prüfung der Roboterschule durchgefallen ist, gibt ihm sein Lehrer eine letzte Chance. Auf dem Planeten Odoretta hat sich eine Rasse von kleinen wilden Kobolden niedergelassen, und alles, was bisher tadellos funktionierte, versinkt nun im Chaos.

Daß Sie SWEEVO'S WORLD hier unter den Jump and Run Spielen finden, liegt ganz einfach daran, daß neben der reinen Betätigung als Abenteuertüftler ein ganzes Stück Arcaden- und Sprunggeschick dazugehört, um hierbei auf einen grünen Zweig zu kommen.

Die Grafik ist zweifelsohne von Ultimate inspiriert, aber nicht so gelungen in Szene gesetzt, wie das dort geschehen ist. SWEEVO'S WORLD ist ein verdammt schweres Spiel. Stellenweise sind dort exakt getimte und pixelgenau auszuführende Sprünge notwendig, um an das gesteckte Ziel zu kommen. Abgesehen davon, daß die Motivation bei sol-


Boulder Dash



Hanse

chen, meist mißlingenden Aktionen auf der Strecke bleibt, ist SWEEVO'S WORLD ein Beispiel weniger gelungener Spiele, an dem der Spieler durchaus verzweifeln kann.

BOULDER DASH: Rockford liebt Diamanten. Außerdem weiß er, wo eine ganze Menge von den Kleinodien herumliegen. Das Problem bei der Sache ist nur, daß die Höhle, in der die Diamanten herumliegen, eine echt wackelige Angelegenheit ist. Stößt Rockford auch nur einmal an den falschen Stein, kann es ihm passieren, daß sich ein Felsbrocken löst und ihm auf den Kopf fällt. Damit ist Rockford dann auch eines seiner Leben los.

Außer den gefährlichen Felsen gibt es auch noch die sonderbarsten Wesen, die sich in dem Höhlensystem herumtreiben und auf Opfer warten. Aber Rockford ist clever, er paßt die Untiere an den richtigen Stellen ab und läßt ihnen Steine auf den Kopf fallen. Einige dieser Wesen verwandeln sich daraufhin in einen Berg Diamanten, andere hingegen lösen sich in Luft auf, wenn sie von einem der Steine getroffen werden.

So manche der 12 verschiedenen Level, die jeweils aus einer sehr großen, gescrollten Spielfläche, die die Höhle, in der Rockford sich gerade befindet, als Querschnitt zeigt, ist eine echte Tüftelaufgabe für sich. Stellenweise gibt es nur einen einzigen gangbaren Weg, um ans Ziel, das heißt in den nächsten Level, zu kommen.
1985 wurde BOULDER DASH von einer amerikanischen Computerzeitschrift zum Spiel des Jahres gewählt. Das hat es auch, ganz ehrlich, verdient.

Inzwischen gibt es zwar schon zwei Fortsetzungen, die in der Machart dem ersten Spiel gleichen, aber das erste ist immer noch das beste. Ein echtes „Einsame-Inseln-Spiel".

Simulationen

HANSE: Wirtschaftsspiele haben, spätestens seitdem es Monopoly gibt, einen festen Platz auf dem Spielemarkt. Und was würde sich besser dazu eignen, zu einem Computerspiel gemacht zu werden, als ein Spiel, bei dem es darum geht, Geschäfte, die haarscharf kalkuliert werden müssen, abzuschließen. Und ums Handeltreiben geht es bei HANSE dann auch im wesentlichen.

Angesiedelt ist die Geschichte um das Jahr 1200 n.Chr. in der wohlhabenden Hansestadt Lübeck. Als junger Geschäftsmann müssen Sie versuchen, durch geschickten Wareneinkauf und die ebenso wichtige Verschiffung der Güter, reich zu werden. Dabei ist ein gut geplantes Verschiffen der Güter ebenso wichtig, wie der günstige Einkauf, gilt es doch herauszufinden, wo das gerade geladene Gut den besten Preis erzielt. Wer Waren hortet und in seine Lagerhäuser stellt, auf daß sie bald teurer werden, kann böse Überraschungen erleben. Ratten und Fäule waren zu jenen Zeiten weit verbreitet. Wenn die Geschäfte einmal mit besonders harten Bandagen geführt werden müssen, dann besteht auch die Möglichkeit, einen Händel anzuzetteln, was in etwa einem klitzekleinen Krieg entspricht.

Hanse ist wie ein Gesellschaftsspiel aufgebaut, an dem sich bis zu vier Spieler beteiligen können. Ein einzelner Spieler tritt gegen den Computer an. HANSE ist nicht schlecht gemacht, und bringt, wie sollte es anders sein, mit mehreren Spielern den meisten Spaß.

Fazit

Wie man sieht, ist 1986 ein sehr interessantes Jahr gewesen, und sicherlich werden Sie das eine oder andere Spiel in dieser Aufzählung vermissen.
Gerne hätten wir Ihnen noch viel, viel mehr Spiele vorgestellt, was leider allein aus technischen Gründen nicht möglich war. Trotzdem hoffe ich, daß Ihnen unser kleiner Streifzug durch die Softwareszene gefallen hat.

(HS)





DAS ERSTE COMPUTERMAGAZIN FÜR HOCHSPRACHEN

PASCAL

INTERNATIONAL

11/86

Oktober
November
1986
1. Jahrgang
DMV

Multi-Plan-Datei-Zugriff
GRAPHISCHE DARSTELLUNG LEICHT GEMACHT

Datenstrukturen mit Zeigern

Pascal-Review
IMPLEMENTATION AUF DEM ATARI ST

Praxis-Bibliothek
KURVENDISKUSSION

Interview mit Fritz Heimsoeth
EIN MARKETINGKONZEPT SETZT SICH DURCH



PASCAL

Pascal - DIE ANDERE - Computerzeitschrift
ab 15.10. bei Ihrem
Zeitschriftenhändler

Adventures à la Creme de la Creme



Amerika im Jahre 2031. Tendenzen, die sich schon in unseren Tagen abzeichnen, haben sich bestätigt. Kriminalität und soziale Ungerechtigkeit fanden kein Ende, vielmehr haben sie sich noch weiter verschlimmert und sind für die Betroffenen fast unerträglich geworden. Rassismus und Kriegshetzerei befinden sich auf dem Vormarsch. In den Vereinigten Staaten gibt es ganze Landstriche, in denen die Armut immer mehr um sich greift. Auch der verbannte Hungertod findet in den Staaten wieder Opfer, zumeist unter den Ärmsten und Schwächsten. Anderswo auf der Welt sieht es nicht viel besser aus, eher sogar noch schlimmer.

In den Entwicklungsländern hat durch anhaltende Wasserknappheit die schlimmste Hungerkatastrophe der letzten zweihundert Jahre eingesetzt. Die Situation steuert unaufhaltsam in Richtung einer globalen Katastrophe.

Wieder Amerika des Jahres 2031, Rockville, Dakota. Eine kleine Stadt, die hauptsächlich durch das ganz in der Nähe liegende Mount Rushmore Nationaldenkmal bekannt wurde. In dieser Stadt hat die PRISM Projektgruppe ihr Domizil aufgeschlagen. PRISM ist eine Großrechenanlage, die speziell für Simulationsmodelle entwickelt wurde. Die brisante Situation, in der sich Amerika und der Rest der Welt befindet, ist der Wissenschaft kein Geheimnis geblieben. Mit PRISM sollen mögliche Maßnahmen, die alles wieder zum Guten wenden sollen, mittels der Rechenanlage simuliert und analysiert werden. Zu diesem Zwecke wurde PRISM entwickelt. Aber PRISM ist eigentlich nicht nur ein Rechner, der Situationen und mögliche Entwicklungen simulieren kann, sondern beherbergt ebenfalls eine Persönlichkeit, die eigentlich ihrerseits nichts anderes als ein besonderes Programm ist, ein Programm, das denken, fühlen und überlegen kann. Sozusagen eine hundertprozentige künstliche Intelligenz, die mit einer solchen Akribie entworfen wurde, daß sie sich selbst zwar darüber im klaren ist, daß sie eigentlich ein Rechner ist, sich selbst aber trotzdem als ein Individuum fühlt.

Dieses digitale Wesen ist eine der wesentlichen Komponenten des PRISM-Programmes. Während auf einer anderen Anlage die Welt nach einer durchgreifenden politischen Maßnahme simuliert wird, linkt sich die Persönlichkeitssimulation in das so entstandene Szenario ein. Seine Aufgabe dort ist es, wesentliche Vorgänge aufzuzeichnen – zum Beispiel eine schiedlichsten Neuigkeiten zurückzugreifen. Der Spieler, der sich "AMFV" zum ersten Male anschaut, wird von der Vielzahl der verschiedenen Betriebsmodi sicherlich ein wenig verwirrt sein. Mir ging es da nicht viel anders. Aber es ist schon eine echte Umstellung vom Leben, Denken und Fühlen eines einfachen Menschen auf die inmobile Existenzebene und die dementsprechend unterschiedliche Seelenlandschaft einer denkenden Maschine.

Ebenfalls interessant sind die Giveaways, die üblicherweise jeder Infocom-Spielpackung beiliegen, und obwohl es sich immer nur um interessant gestaltete Kleinigkeiten handelt, sind diese immer ein wesentlicher Bestandteil des eigentlichen Spiels. Bei "AMVF" stehen dem Spieler ein Stadtplan von Rockville Dakota, ein Werbekugelschreiber einer örtlichen Skycar-Vermietung, und eine, wie eine Rechenscheibe angelegte PRISM-Codekarte zur Verfügung.

Der Stadtplan erleichtert die Orientierung während der eigentlichen Simulationsphase ungemein. Die PRISM-Codekarte dient zur Ermittlung des jeweils aktuellen Zugangscodes für eine Simulation. Der eigentliche Zweck dieser Karte, läßt sich erst erkennen, wenn man weiß, daß Infocom-Adventures in der Regel nicht geschützt sind, sich im Grunde genommen also problemlos kopieren lassen. Eine solche Kopie ist auch tadellos lauffähig, aber ohne die Codecarte nicht spielbar. Denn bereits beim ersten Einklinken in eine Simulation erwarten die Sicherheitssysteme von PRISM (und damit Infocom) die Eingabe eines Codes, der sich nur mittels der Karte herausfinden läßt. Und auch den hartgesottensten Schwarzkopierern sei sogleich der Wind aus den Segeln genommen – die beschriebene Karte läßt sich auch mit dem besten Fotokopierer nicht duplizieren.

Zork

Während "A Mind forever Voiaging" eines der neuesten Infocom-Produkte ist und in einer feinen Aufmachung und Ausführung in die Hände des Spielers gelangt, möchte ich Ihnen hier nun die Spiele vorstellen, die im wesentlichen den guten Namen der Infocom-Leute begründet haben.

Zu einer Zeit, als alle Hersteller – koste es was es wolle – versuchten, alle ihre Spiele und Adventures mit Grafik zu versehen, um auf diese Art und Weise in der Anwendergunst zu Gemeinderatssitzung, oder ein Gespräch mit einem Priester. Anhand dieser Aufzeichnungen und deren Analyse, gedenken die Wissenschaftler einen möglichen gangbaren Weg aus der Misere zu finden. Daß dies nicht nur in Bezug auf Technologie und Programmierung einiges an Problemen aufwirft, stellt sich heraus, als ein konservativer Senator versucht, das PRISM-Projekt zu stoppen. Er verteufelt in seinen Reden die Arbeit der Wissenschaftler und es gelingt ihm, auf diese Art und Weise ein Stimmungsklima zu schaffen, in dem die Arbeit für das PRISM-Team fast unmöglich wird. Die einzige Möglichkeit, dieser Kampfansage zu begegnen, ist das schnelle Erzielen von handfesten Erfolgen.

Dies ist in groben Zügen die Handlung eines der neueren Infocom Adventures. Sollten Sie sich schon das eine oder andere Mal mit einem Abenteuer beschäftigt haben, wird Ihnen sicherlich aufgefallen sein, daß die Story, auf die dieses Spiel aufbaut, mehr als unüblich für das Genre ist. Ranken sich die Storylines üblicherweise um entführte Prinzessinen oder verborgene Schätze, erinnert "A Mind forever Voyaging" eher an Orwell's 1984 oder Huxleys "Schöne neue Welt". Zweifelsohne fühlt sich "AMFV" in dieser Gesellschaft recht wohl, denn erklärtes Ziel der Programmierer war es, eine Social Fiction Story in der Infocom eigenen Art und Weise in ein interaktives Adventure einzubauen.

Auch dieses Adventure zeichnet sich durch die außerordentlich hohe Qualität der Texte aus, aber das gehört bei Infocom ja fast schon zum guten Ton. Aber trotzdem ist es immer wieder erstaunlich, was dem Spieler in den Infocom-Welten alles begegnen und passieren kann. Im vorliegenden Falle ist es gelungen, die pessimistische Stimmung, die überall vorherrscht, hervorragend darzustellen. Der Spieler, der sich mit diesem Programm beschäftigt, findet sich unversehens in der Rolle der PRISM-Persönlichkeit wieder. Da diese Wesenseinheit, wie beschrieben, lediglich eine Computersimulation ist, kann sie sich natürlich nicht frei bewegen wie ein normaler Mensch, zumindest nicht außerhalb der Simulationen. Zwar hat das PRISM-Bewußtsein Zugang zu allen Videokanälen des Institutes, aber Informationen über Vorgänge außerhalb des Gebäudes sind ihm nur über die verschiedenen Netzwerkrechner zugänglich, an die es angeschlossen ist. So besteht die Möglichkeit, vom Wetter bis zu den Nachrichten, auf die untersteigen, bewiesen die maßgeblichen Leute bei Infocom einigen Mut, und entschlossen sich dazu, den allgemeinen Markttendenzen zum Trotze, ihre geplanten Spiele als reine Textadventures zu verwirklichen.

Von vorneherein war klar, daß, wenn man schon auf die verkaufsträchtige Grafik verzichtet, das Programm an sich bis ins letzte Detail stimmig und intelligent durchdacht sein mußte, um damit zu landen. Da sich zu diesem Zeitpunkt die Fantasywelle gerade auf ihrem Höhepunkt befand,

gab es auch kein langes Federlesen um das Thema des Spieles. Etwas Fantasyartiges mußte es schon sein. Es wurde begonnen einen Plot auszuarbeiten, der sich dafür eignete, in ein Abenteuerspiel umgesetzt zu werden, und nach und nach nahm die Geschichte Gestalt an.
Angesiedelt wurde die Story in einem unglaublich großen unterirdischen Gängesystem, dem großen Untergrund-Königreich. Der Ursprung dieses Tunnelimperiums liegt sicherlich auch daran, daß sich ein Abenteuerspiel in den fest umrissenen Örtlichkeiten eines Höhlensystems um einiges besser ansiedeln läßt, als eine „Wald und Wiesen Story", bei der nach keiner Seite eine Grenze ist, die dem Spieler vermittelt, daß es in dieser Richtung nichts Interessantes mehr zu erkunden gibt. Ein Umstand, der bei so manchem anderen Adventure-Spiel bisweilen ein wenig sauer aufstößt.
Der Hintergrund der Story rankt sich im wesentlichen um die Entstehung dieses großen Untergrund-Königreiches, das ein einigermaßen verrückter König, nachdem er ein ähnliches Reich, allerdings in einem natürlichen Höhlensystem errichtet, gesehen hat, durch seine Architekten und Steinmetze in einen großen Berg schlagen ließ. Als er dann allerdings mit seinen Untertanen in sein neues Reich umziehen wollte, weigerten sich die meisten von vorneherein, und die, die es erst einmal ausprobieren wollten, ließen es schnell bleiben und suchten das Weite. So kam es, daß sich das große Untergrundimperium nach und nach selbst entvölkerte, und schon bald lebten nur noch der König und seine engsten Vertrauten in den dunklen Gewölben. Zu dem Zeitpunkt, an dem die eigentliche Spielhandlung beginnt, sind diese Ereignisse schon lange zur Geschichte geworden, über die nur noch das eine oder andere Schulbuch zu berichten weiß.

Sieht man sich ZORK I einmal genauer an, wird schnell klar, daß die Programmierer und Designer von Infocom sich die ganze Angelegenheit sehr genau überlegt haben. Wesentlicher Schwerpunkt lag hierbei zweifelsohne auf der Erstellung der Texte, die dem Spieler im Laufe seiner Aktionen via Monitor präsentiert werden.
Deutlich tritt hier auch erstmals der unnachahmliche Infocom-Witz zu Tage, über den man allenthalben stolpert, wenn man sich an die Erkundung des Höhlensystem macht. So steuert bereits die gedruckte Einleitung eine recht humorvolle Erklärung für den Bau des Untergrundimperums bei. Das sich der verrückte König am Anfang seiner Karriere sehr für die Kriegskunst interessierte und auch noch ein wenig eroberungslüstern war, überfiel er alle seine Nachbarkönigreiche und eroberte dieselben. Dabei fielen ihm auch die Corn Flakes Minen einer benachbarten Monarchie in die Hände. Leider mochte keiner seiner Untertanen Corn Flakes, und er stand mit Tonnen des Getreideproduktes da, und wußte nicht, wohin mit dem Zeug. Auf der Suche nach einem geeigneten Absatzmarkt für seine Corn Flakes segelte er auf der großen See die Kreuz und die Quer. Auf einer dieser Fahrten entdeckte er dann auch das natürliche Untergrundimperium, das in derart nachhaltig beeinflußte.

Während in ZORK I die Erkundung des Höhlenlabyrinthes im Vordergrund steht, wird der Handlungsrahmen in ZORK II schon um einiges komplexer. Nun geht es darum, einen finsteren Zauberer unschädlich zu machen, der sich allerdings irgendwo in einem sehr abgelegenen und lange vergessenen Teil des unterirdischen Komplexes versteckt hält. Auf der Suche nach diesem Magier gelangt der Spieler durch Regionen, die belegen, daß sich der König, der das Untergrundimperium auf dem Gewissen hat, einige Gedanken um das Wohlergehen seiner Bevölkerung gemacht hat. So gelangt man beispielsweise in eine gewaltige Kaverne, die sich Grayslopes nennt. Dort war einmal, als es noch Bewohner gab, das Wintersportzentrum des unterirdischen Königreiches. Wer sich mit den winterlichen Skivergnügungen nicht so recht anfreunden kann, dem sei der Bozbarland Vergnügungspark empfohlen. Dort findet der vergnügungssüchtige Abenteurer über 200 verschiedene Attraktionen, die vom einfachen Glücksspiel bis zur gestandenen Geisterbahn reichen. Allerdings sollte man sich den Besuch der Geisterbahn zweimal überlegen, denn die Attraktionen, die sich in diesem Gruselkabinett befinden, sind zumeist Dämonen und Ungeheuer in Natura. Eine Attraktion, die man, so verspricht es der Prospekt, niemals wieder vergißt, sofern man sie überlebt.
Nach und nach bekommt der Spieler im Laufe von ZORK I + II immer wieder Hinweise auf eine eigenartige Person, dem sagenumwobenen Höhlenmeister des Untergrundimperiums, der, obwohl so viele Jahre der tödlichen Stille auf den Kavernen und Höhlen lasten, noch immer seine Ränke schmiedet und für die Erhaltung des Imperiums sorgt. ZORK III handelt dann auch, wie Sie sich sicherlich schon gedacht haben, von der Suche und dem Unschädlichmachen dieses legendenumrankten Mannes.

Das Meiste, was auf diesen Seiten über Infocom und deren Produkte gesagt wurde, trifft, natürlich mit kleinen Einschränkungen, auch auf die anderen Infocom-Adventures zu. Die besondere Art wie das Spiel mittels einiger geschickt ausgewählter kleiner Gegenstände einen realistischen Anspruch erhält. Gerade diese "Add Ones" vermitteln dem Spieler eine Authentizität, die in dieser Form von keinem anderen Adventure erreicht wird.

Leider war es bisher so, daß alle CPC-Besitzer auf den Spaß ein solches interaktives Abenteuer zu spielen, verzichten mußten. Aber das soll anders werden. Seitdem Infocom durch Activison übernommen wurde, plante man eine Umsetzung für die CPC's, die, unter CP/M laufend, endlich auch auf den Amstrad-Rechnern deren Anwendern interaktive Abenteuerfreuden bescheren sollen. Die Realisierung dieses mit nicht unbeträchtlich viel Arbeit verbundenen Projektes, steht kurz vor ihrem Abschluß. Allen Abenteurern kann in diesem Zusammenhang nur empfohlen werden, sich einmal an einem solchen Spiel zu versuchen. Möglicherweise übernimmt Activision ja auch ein kleines Feature mittels dessen Infocom sich eine ganze Reihe guter Freunde gemacht hat. In den Vereinigten Staaten gab es nämlich eine Zeit lang den sogenannten 'Starter Pack". Dies war ein Paket von drei verschiedenen Adventures, die zu einem ausgesprochen günstigen Preis unter die Leute gebracht wurden. Eine für den Kunden wie auch für Infocom interessante Aktion, war es doch damit möglich, sich für verhältnismäßig wenig Geld mit der Abenteuerwelt der interaktiven Novellen vertraut zu machen. Auch Infocom kommt bei diesem Geschäft nicht zu kurz, zeigt doch die Statistik das fast neunzig Prozent der Anwender, die sich einmal ein Infocom Abenteuer zugelegt haben, sich danach noch weitere Spiele zulegen. Es verhält sich hier wirklich so, daß, hat man erst einmal Blut geleckt, so richtig auf den Geschmack kommt.

Verraten und Verkauft

Wer sich schon einmal an einem Computerspiel versucht hat, kann sicherlich, wie viele andere vor ihm schon, ein Lied über mangelhafte Bedienungsanleitungen singen. Ob es nun die deutsche Übersetzung einer englischen Anleitung ist, oder eine, die erst gar nicht übersetzt wurde. Bisweilen kann man sich über derartige Übersetzungen, die meistens von englischen Mitarbeitern der einzelnen Softwarehäuser unter Zuhilfenahme eines Wörterbuches erstellt wurden, köstlich amüsieren. Die Grammatik, die im Zuge solcher Aktionen zu Papier gebracht wird, ist in den meisten Fällen schlicht und ergreifend als verheerend zu bezeichnen.

Trotzdem ist dieser Punkt nicht der drastischste Mangel, dem man in solchen Fällen begegnet. Denn selbst mit einer unmöglich geschriebenen Anleitung, die zumindest inhaltlich richtig ist, läßt sich mit ein wenig Phantasie arbeiten. Aber was, wenn von vornherein gar keine Anleitung beiliegt, oder sich die Anleitung als völlig verkehrt erweist?
Dann fängt das Rätselraten an. Besonders hartnäckige Naturen werden sicherlich versuchen, sich im nachhinein eine Bedienungsanleitung von dem Händler zu besorgen, bei dem sie auch das Spiel erwarben. Mitunter führt dies auch zu beachtlichen Erfolgen, aber nicht immer ist der Händler in der Lage, einem solchen durchaus berechtigten Wunsch Rechnung zu tragen. Denn es kommt vor, daß ein und dasselbe Spiel in mehreren verschiedenen Verpackungen verkauft wird.
So geschehen zum Beispiel bei einem neuen Melbourne House Produkt; die Kassetten-Version von Rock'n Wrestle lag uns in zwei verschiedenen Verpackungen zum Test vor. Zum einen in einem Doppel-Kassetten-Pack, zum anderen in einer kleinen, nur für eine Kassette ausgelegten Plastikhülle. Erstaunlicherweise lag der großen Packung lediglich das Titelinlay bei, von einer Bedienungsanleitung nicht die Spur. In die kleinere Kassette hingegen hat der Hersteller dann mit Brachialgewalt noch eine Anleitung zugelegt. Die Hülle war dadurch so überfüllt, daß, um die Kassette und die Anleitung aus der Hülle zu bekommen, ein mittlerer Gewaltakt vonnöten war. Daß der großen Verpackung keinerlei Bedienungshinweise beilagen, ist nicht, wie man meinen könnte, ein Einzelfall; viele unserer Leser erwarben dieses Spiel und wurden nach dem Öffnen den Mißstand gewahr.
Was bleibt dem frustrierten Anwender, nachdem die Ausschöpfung aller seiner Möglichkeiten, im nachhinein einer Anleitung habhaft zu werden, keinerlei Ergebnisse brachten? Er ist, wie auch zu anderen Gelegenheiten im Leben, auf sich selbst gestellt.
Zu allererst sollte er das Programm einmal in den Rechner einladen, denn in einigen Fällen ist die Anleitung ein Bestandteil des Programmes selbst - ist dies nicht der Fall, sollte umgehend Papier und Bleistift ins Spiel gebracht werden. Unter Zuhilfenahme der erwähnten Accessiors geht es jetzt daran, sich selbst die fehlende oder mißratene Anleitung zu erstellen.
Liegen Papier und Bleistift (im Notfall kann auch auf einen Kugelschreiber zurückgegriffen werden, Heizkörperpinsel und Deckenrollen sollten vermieden werden, sie haben in unserem Test nur sehr mangelhafte Ergebnisse erbracht.) bereit, ist das Programm fehlerlos im Speicher des Rechners und läuft ohne Beanstandungen, kann mit der empirischen Erarbeitung des benötigten Papiers begonnen werden.
Das Groß der Spielprogramme hat die Angewohnheit, sich zuallererst mit einem Titelbild zu melden. Dieses Titelbild sollte ohne Zögern auf etwa darin verborgene Hinweise untersucht werden. Wenn Ihre Suche nach derartigen Mitteilungen von Erfolg gekrönt worden sein sollte, versuchen Sie sich zu vergegenwärtigen, was diese kryptischen Schriftzeichen bedeuten könnten. Sollte es der Fall sein, daß das Szenario, das sich vor Ihnen auf dem Monitor präsentiert, ein klassisches Beispiel für ein sogenanntes Menue ist, erkunden Sie, notfalls unter Zuhilfenahme eines Wörterbuches, die Bedeutungen der einzelnen Menuepunkte. Im allgemeinen dienen derartige Menues zum Einstellen diverser Optionen, sollten Sie also bei Ihrer Suche nach einem derartigen Menue fündig geworden sein, können Sie nun damit anfangen, das Programm nach den von Ihnen gewünschten Gegebenheiten einzustellen. Im Zuge dieser Anpassung sollte es Ihnen eigentlich nicht allzu schwer fallen, die von Ihnen gewünschte Steuerung (Tastatur, Joystick, Fahrradlenker), und die Anzahl der Spieler (Single, Ehepaar, Fischerchöre) einzustellen. Des weiteren enthält ein solches Menue, meistens an dessen Ende plaziert, eine Option, mit der sich - oh Wunder - das eigentliche Spiel starten läßt.
Sollte es sich um ein Spiel handeln, bei dem der Hersteller es vermieden hat, auf der Packung einige Szenenfotos abzudrucken, kommt nun der große Moment, in dem sich erweist, wie schlecht das Programm nun wirklich ist. Ein Naturgesetz der Softwareszene besagt nämlich, daß mindestens die Hälfte aller ehrlich erworbenen Programme einer Softwaresammlung schlecht oder zumindest nicht gerade sehenswert sind.
Ein gutes Rezept für Personen, die sich ein neues Spiel erworben haben, welches sich als eines der weniger guten entpuppt: Eine Multiplikation der Begeisterungsfähigkeit, in Bezug auf ein schlechtes Spiel, hat ihrerseits eine subjektive Steigerung der Spielqualität zur Folge. Sollte auch das nicht wirken, so besteht immer noch die Möglichkeit, mit ein paar Bier nachzuhelfen, oder in ganz hoffnungslosen Fällen, das Spiel ganz einfach in den Müll zu werfen.
Nun, nachdem Sie den Schock, resultierend aus der Diskrepanz zwischen der Coverillustration und der Grafik des eigentlichen Spiels hoffentlich gut überstanden haben, gelangen Sie in das eigentliche Spiel. Hier erweist sich dann auch, um welche Art von Spiel es sich handelt.
Testhalber sollten Sie an dieser Stelle einmal mit dem Joystick alle möglichen Bewegungen ausprobieren und gleichzeitig aufmerksam den Monitor beobachten, bewegt sich dort ein Objekt oder eine Figur simultan zu Ihren Joystickbewegungen, so läßt sich daraus schließen, daß dies die vom Spieler zu steuernde Figur ist. Bewegt das Männlein oder Objekt, oder was immer es sonst darstellen soll, sich in alle vier Himmelsrichtungen, legt das den Schluß nahe, daß es sich bei dem Spiel um eines mit einem Draufsicht-Szenario handelt. Marschiert der kleine Pixel-Paul allerdings lediglich nach links und rechts, bedeutet dies meistens, daß hier ein Querschnitt als Szenario dient. Betätigen Sie nun einmal kurz den Feuerknopf (nicht den am Feuerzeug), springt das Bildschirmwesen

daraufhin in die Luft, so ist dies ein weiteres Indiz dafür, daß es sich hier um ein sogenanntes Jump & Run Spiel handelt.

Jump and Run forever

Seitdem weiland ein gewisser Manic Miner durch eine verschollene Mine stolperte und Donkey Kong, seines Zeichens King Kong's kleiner Bruder, dem armen Mario immer neue Fässer in den Weg warf, erfreut sich diese Spielegattung eines mehr oder weniger ungebrochenen Interesses. Grundsätzliches Ziel solcher Spiele ist es, das Spielfigürchen durch ein mit den teuflischsten Todesfallen gewürztes Gerüst zu manövrieren.
Erschwert wird das ohnehin schon nicht leichte Vorankommen durch diverse umherfliegende oder marschierende Untiere, Gegenstände oder andere Dinge, deren einzige Existenzberechtigung darin besteht, dem Spieler das Leben so schwer wie möglich zu machen.
An dieser Stelle kommt das Timing ins Spiel, denn meistens ist es bei solchen Spielen unmöglich, ohne das richtige Timing auch nur die erste der Plattformen oder Leitern zu erreichen. In der Regel ist man dem Punkt, an dem das Spiel beginnt oder in ein neues Bild eintritt, sicher. Die fliegenden Widersacher meiden diesen Bereich, aber auch in diesem Falle gilt: Ausnahmen bestätigen die Regel.
Der Begriff „Timing" meint im genannten Zusammenhang, daß der Spieler den Moment auskundschaften muß, der am geeignetsten ist, um sich an den feststehenden und den beweglichen Hindernissen vorbeizustehlen, ohne dabei ein Leben zu verlieren. Dieses Timing wird in den meisten Fällen durch die unmöglichen Bewegungsrythmen der Widersacher bestimmt.
Diese Rythmen sind so angeordnet, daß das Springen und Laufen bis auf einen Pixel genau erfolgen muß. Ist die Spielfigur zu schnell oder zu langsam, hilft auch kein Glück mehr, es kommt unweigerlich zur Kollision und damit zum Verlust eines Lebens. Also, zuerst einmal alle Unholde genau beobachten, um auf diese Art und Weise die einzig gangbare Route zu erkunden.

Die Arcaden-Tortur

Ein entfernter Verwandter der Jump and Run Spiele ist das sogenannte Arcaden-Adventure. Diese sind in der Regel gut daran zu erkennen, daß sich das Bildschirmdisplay in der Draufsicht präsentiert. Da es in der Branche sozusagen üblich ist, ein Spiel mit einer kleinen Vorgeschichte zu versehen, und diese Geschichten in 99 % aller Fälle fast völlig identisch sind, möchte ich es nicht versäumen, Ihnen an dieser Stelle die Standard-Hintergrundstory der Arcaden-Adventures zu erzählen.
Es war einmal ein böser Zauberer (König, Priester, Limoverkäufer), dessen Herz sich am Anblick einer schönen Prinzessin (Burgfreulein, Jungfrau, Putzfrau) in Liebe entzündete. Allerdings gab es da noch einen tapferen Recken (Ritter, Prinz, Hausmeister), der schon seit einiger Zeit mit der schönen Maid verlobt war, und dem des Zauberers Ambitionen gar nicht gefielen. Auch unsere Heldin wies das Werben des Finsterlings ab. Daraufhin erzürnte dieser sich fürchterlich und entführte die Prinzessin auf seine schwarze Burg (Turm, Höhle, Trinkhalle). Das machte unseren Helden allerdings erst so richtig wütend und er brach auf, um dem Finsterling den Garaus zu machen.
Dieser hatte sich zu diesem Zeitpunkt bereits davon überzeugt, daß der Held auf dem Wege sei und lotst ihn in ein unterirdisches Labyrinth (Gruft, Keller, U-Bahnhof), um dort alle seine Monster (Ungeheuer, Wesen, Fans) auf ihn zu hetzen.
Nun, soweit so gut, eigentlich wollte ich dieses Thema gar nicht so sehr vertiefen. Um bei der Story zu bleiben, das eigentliche Spiel beginnt mit dem Eindringen des Helden in den unmittelbaren Machtbereich des Bösewichtes. Dort gilt es, eine unergründliche Vielfalt von Gängen und Zimmern, in denen sich, wie bereits erwähnt, die Untiere des Chef-Finsterlings herumtreiben, zu durchwandern. Diese Monstren lungern überall in den Gewölben herum und warten auf ihr Mittagessen - äh, pardon - den Helden, um ihn gebührend zu empfangen.
Im wesentlichen fangen wir bei einem solchen Spiel ebenso an wie der Vorgang weiter oben bereits beschrieben wurde. Ist das Menue mit all seinen Fährnissen und Klippen umschifft, beginnt auch hier das eigentliche Spiel. Ein grundsätzlicher Unterschied wird schon in den Grundzügen klar. Während bei einem Jump and Run Spiel bereits eine Berührung mit einem Widersacher ausreicht, um eines seiner Leben zu verlieren, verhält es sich bei den Arcaden-Adventures meistens so, daß dem Helden lediglich ein Leben zur Verfügung steht. Dieses Leben wird durch sogenannte Lebenspunkte dargestellt, das heißt, jede Berührung mit einem Widersacher führt auf der Lebenspunkteskala zu einem Punktabzug, und ist man bei Null angelangt, so ist damit auch das Spiel vorbei.

In der Regel verhält es sich so, daß im Labyrinth verstreut immer wieder Gegenstände auftauchen, deren Einsammeln eine Steigerung der Lebensenergie bewirkt.
Timing ist bei dieser Art von Spiel zwar ebenso notwendig wie bei den Jump and Run Spielen, aber bei weitem kein so wesentliches Moment wie dort. Vielmehr kommt es hier darauf an, die kleinen, lebensspendenden Leckereien richtig einzuteilen und, was ganz besonders wichtig ist, sich nicht zu verirren. In solchen Fällen hat sich eine selbst angefertigte Karte stets als sehr nützlich erwiesen. In eine solche Karte sollten auch alle Gegenstände, die sich in den einzelnen Räumen befinden, eingetragen werden. Auch sollten Sie nicht vergessen anzumerken, ob die Gegenstände beim Aufnehmen positive oder negative Auswirkungen haben. Sie werden feststellen, daß Sie sich, nachdem Sie die Karte erst einmal gezeichnet haben, auch recht gut ohne diese klar kommen, alleine das Durchwandern und Aufzeichnen reicht aus, um sich alles Wesentliche genau zu merken. Und dies ist eine Grundvoraussetzung für eine 100 %-ige Lösung aller Probleme des Spieles.
Aber noch ein anderes Moment unterscheidet die Arcaden-Adventures von anderen Computerspielen, nämlich das gewisse Gegenstände oder Wege auf den ersten Blick nicht sichtbar sind. Den einzigen Tip den ich Ihnen in diesem Zusammenhang geben kann ist: Untersuchen Sie alles, nehmen Sie jeden Gegenstand auf, versuchen Sie herauszubekommen, ob er mit anderen Gegenständen zusammen eingesetzt werden muß. Sehen Sie sich jeden Raum auf das genaueste an, es besteht immer die Möglichkeit, daß ein ganz spezielles Ding an einen ganz speziellen Ort gebracht werden muß, und dort erst seine wahre Natur offenbart.

Kung Fu, Catchen und Mikado

Eine weitere Spielgattung findet gerade in den letzten Monaten immer mehr begeisterte Anhänger. Die sogenannten Martial-Arts sind schon des

öfteren zum Thema von mehr oder weniger guten Spielen gemacht worden. Meist reicht es hier, den Computergegner so heftig wie möglich aufs Haupt zu hauen, um ihn auf dem Boden der Computerarena niederzustrecken.

Diese Spiele gibt es, obgleich der Spielgedanke noch recht neu ist, bereits in den unterschiedlichsten Ausführungen. Eines aber ist ihnen allen, ob Arena oder Urwald, gemein, sie sind mittels eines Joysticks ungemein leichter zu handhaben, als dies via Tastatur möglich wäre. Sollte dieser nicht vorhanden sein, bleibt Ihnen wohl oder übel nichts anderes übrig, als in den sauren Apfel zu beißen, und sich auf der Tastatur Knoten in den Fingern zu holen.

Die Steuerung solcher Kampfsportsimulationen ist im Grunde genommen recht simpel. Es stehen mehrere verschiedene Aktionen zur Verfügung, mit deren Hilfe der Spieler angreifen oder sich verteidigen kann.

Der Joystickbenutzer, der hoffentlich schon das obligatorische Blatt Papier und einen Stift gut erreichbar neben dem Computer plaziert hat, kann die einzelnen Steuerungsmöglichkeiten im Grunde genommen recht einfach erkunden. Er bewegt den Stick einmal in jede mögliche Richtung (dabei sollten die Diagonalen nicht übersehen werden), ist diese Prozedur abgeschlossen, wird sie einfach mit gedrücktem Feuerknopf wiederholt. Beobachtet man im Laufe dieser Aktion seinen Bildschirmkämpfer genau und notiert sich sogleich, welche Verrenkung das Männchen anläßlich welcher Joystickbewegung macht, ist man im Grunde hinterher um eine Bedienungsanleitung reicher.

Sonst noch was?

Sicherlich gibt es Spiele, die ohne Anleitung geliefert werden und selbst dem härtesten Tüftler auch nach Wochen noch ein komplettes Rätsel bleiben. Am eigenen Leibe mußte ich dies an dem komplett Menuegesteuerten Adventure "Shadowfire" erleben. Dieses Spiel erreichte uns in einer Packung ohne Bedienungsanleitung. Kein Problem, dachte ich so bei mir und lud das Spiel auch sogleich in den Rechner. Aber trotz längerer Anstrengung blieb mir in dieser Sache auch der geringste Erfolg verwehrt. Ich begann, ehrlich gesagt, schon an meinen Fähigkeiten zu zweifeln.

Als dann einige Tage verspätet die Anleitung bei uns im Hause eintraf, entpuppte sich Shadowfire als das Spiel mit dem besten Kopierschutz überhaupt, denn selbst als ich die kleine Broschüre durchgelesen hatte, dauerte es immerhin noch fünf Stunden, bis ich es dann endlich im Griff hatte und sagen konnte: „Jetzt hab ich's kapiert."

Ohne die Anleitung hätte ich mich sicherlich noch bis zum St. Nimmerleinstag mit dem Spiel beschäftigen können, ohne auch nur die Spur von Verständnis für das, was sich da auf dem Monitor abspielte.

Gut, Shadowfire ist ein Extremfall, aber auch bei Spielen, von denen man glaubt, sie wirklich durchschaut zu haben, wird man mitunter immer wieder überrascht. Spezielle Aktionen, die ohne Joystickeinsatz durchgeführt werden und das Betätigen einer bestimmten Tastenkombination am Computer erfordern, lassen sich nur durch Glück herausfinden. Oder aber man findet einen Gleichgesinnten, der ebenfalls das betreffende Spiel erworben hat, aber mehr Glück als Sie hatte, und über das heißbegehrte Papier verfügt. Reden Sie ihm gut zu, vielleicht läßt er mit sich reden und Sie dürfen sich eine Fotokopie machen.

Over und aus

Kaum eine andere Sparte Software kann mit einer derartigen Vielzahl von verschiedenen Produkten aufwarten, wie die Spieleindustrie. Ganz klar, daß ein solcher Ozean aus Unterhaltungssoftware tausenderlei verschiedene Facetten aufweist. Die bisher beschriebenen Genres sind, obgleich sie durchaus zu den vertreterreichsten gehören, nur ein verschwindend kleiner Teil des Ganzen. Viele Spielarten, die unter den abenteuerlichsten Spartenbezeichnungen bekannt sind, konnten hier keinen Platz finden. Die beschriebenen Methoden sind aber bis zu einem gewissen Grad auch auf sie anwendbar.
Gerne hätte ich Ihnen auch zu diesen Spielen spezielle Vorgehensweisen dargelegt, allein der Platz fehlte.
Den besten Tip aber habe ich mir bis zum Schluß aufgehoben. Melden Sie sich bei dem Händler, der Ihnen das Spiel verkauft hat. Teilen Sie ihm den Sachverhalt mit und bestehen Sie darauf, daß er sich darum kümmert, auch für Sie eine Anleitung zu organisieren. Aber vergessen Sie nicht, auch die Händler stellen die Spiele nicht selbst her, und sind somit genau wie Sie, auch der Willkür der Softwareindustrie ausgesetzt.
Absichtlich hat er ihnen das fehlende Papier also sicherlich nicht aus der Packung entwendet. Versuchen Sie zusammen mit ihm das Problem zu lösen. Seien Sie ein emanzipierter Konsument.

(HS)







Abenteuer '86

Das Spielen von Abenteuern ist auch in 1986 immer attraktiver geworden. Auch hat sich die Anzahl der angebotenen Spiele vervielfacht.

Zu absoluten Highlights gesellte sich allerdings auch so mancher echte Flop. Die Themen, die zumeist für Abenteuerspiele herhalten müssen, haben sich kaum verändert. Immer noch stehen SF und Fantasy in der Spielergunst hoch im Kurse, aber es kommt auch hier immer wieder vor, daß sich interessante Spiele mit andersgearteten Themen beschäftigen - bisweilen erst durch ihre von der Regel abweichenden Spielhandlungen an Attraktivität gewinnen und sich aus der grauen Softwaremasse hervorheben.
Eine der interessantesten Möglichkeiten, Abenteuerspiele attraktiver zu gestalten, ist die, in das Spiel interaktive Charaktere einzubauen.

Abenteuern interaktiv

Das Programmieren von reinen Textabenteuern ist, das kann so ziemlich jeder erfahrene Basic-Programmierer bestätigen, im Grunde genommen relativ einfach. So nimmt es auch nicht Wunder, daß sich viele Leute im Laufe ihrer Programmierer-Karriere damit beschäftigen.

Die eigenartigste Eigenschaft solcher Basic-Abenteuer ist, daß sie zwar von der Korrespondenz mit dem Spieler leben, die in der Geschichte auftretenden Personen dies mit bis auf ein fast zum Gestammel abgespeckten Wortschatz allerdings nicht tun. Und damit sind wir bei einen Feature, auf das Basic-Abenteuer fast völlig verzichten müssen - interaktive Charaktere.
Ein interaktiver Charakter, das ist beispielsweise Thorin im "Hobbit" - eine Figur, die vom Computer so gesteuert wird, daß beim Spieler der Eindruck entsteht, daß diese Person über einen richtigen eigenen Willen verfügt. Wenn dieser Charakter, egal ob es sich nun um einen "Er" oder eine "Sie" handelt, gut programmiert wurde, ist es dem Spieler möglich, sich via Keyboard mit dieser Person zu unterhalten. Sicherlich wird dieses Gespräch bei weitem nicht so wie eine Natura-Diskussionsrunde ausfallen, trägt aber enorm zur Atmosphäre des Abenteuers bei.
Egal ob es sich nun um ein einfaches selbsterstelltes Abenteuer oder um ein teures gekauftes handelt, solche Figuren erhöhen die Attraktivität des Spiels und die Motivation des Spielers.

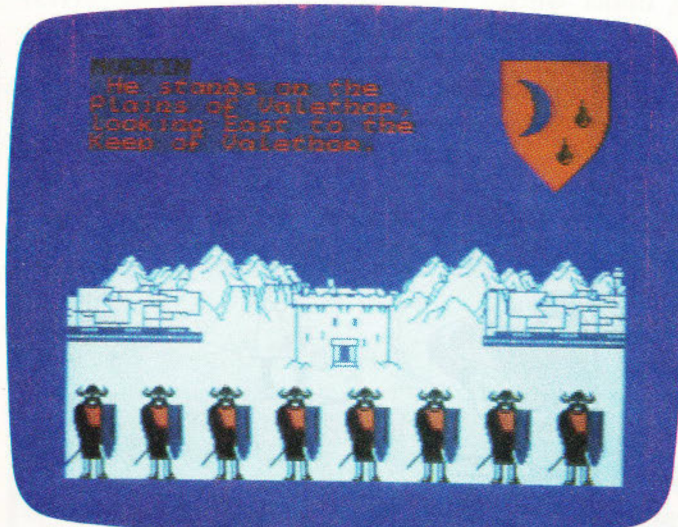
Unglücklicherweise sind die meisten Versuche, solche Charaktere in Programme mit einzubinden, eher schlecht als recht, oder werden, was nur zu oft vorkommt, einfach weggelassen. Die Gründe hierfür werden dem interessierten Beobachter sehr schnell klar. Zum einen verbraucht ein solcher Charakter einiges an Speicherplatz, müssen ihm doch ausreichende Texte zur Seite gestellt werden, durch die er sich dann mit dem Spieler unterhalten kann. Andererseits müssen alle Ortsbeschreibungen so gestaltet sein, daß sie auch über die An- oder Abwesenheit des Charakters Auskunft geben können. Daß sich aus diesen Komponenten dann wiederum eine Vielzahl von Situationen ergibt, die erst einmal durchgetestet werden müssen, versteht sich dann fast von selbst.

Zwar gibt es auch einige Spiele für die CPC's, die gut programmierte interaktive Charaktere beinhalten, der bereits erwähnte Hobbit ist dafür wohl eines der besten Beispiele, im großen und ganzen liegen die interaktiven Charaktere bei den CPC's aber immer noch im Argen.

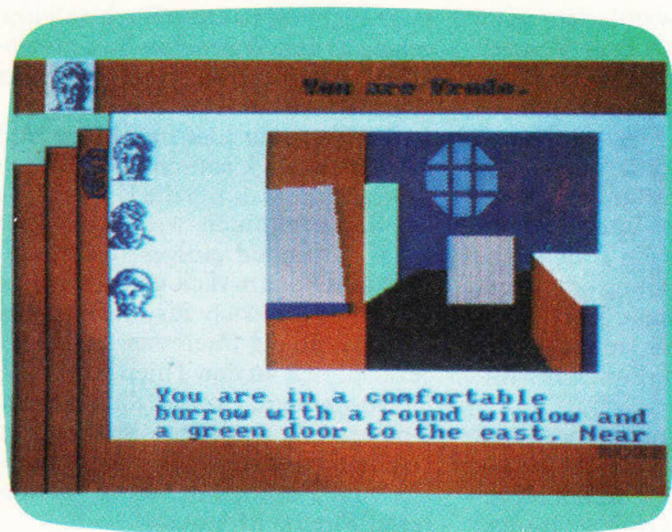
Was ist aber nun ein richtiger interaktiver Charakter, wie definiert man so etwas? Die beste Umschreibung ist meiner Meinung nach folgende: Ein interaktiver Charakter sollte die Möglichkeit haben, nach eigenem Willen im Abenteuerszenario umherzuwandern, er sollte die Möglichkeit haben, Gegenstände nach denselben Gegebenheiten aufnehmen zu können wie der Spieler selbst, und er sollte auf die Dinge, die um ihn herum im Abenteuer stattfinden, eingehen bzw. reagieren.

Abenteuermenue á la carte

Daß das reine Textadventure und das grafikunterstützte Textadventure nicht alles in der Abenteuerwelt darstellten, war klar, gab es doch eine ganze Reihe von sogenannten Arcaden-Adventures - eine Untergruppe der Abenteuer schlechthin, allerdings im großen und ganzen nur durch ein einziges Moment miteinander verwandt, nämlich durch die Tatsache, daß sie in der Regel mit dem Joystick gesteuert werden.
Zwar gibt es diverse Vertreter des Genres, die anstelle eines normalen Joysticks den Cursor-Joystick setzen, aber grundsätzlich bedarf es bei derartigen Spielen auch der Möglichkeit der schnellen Reaktion, und die wird nun mal durch den Stick am besten gewährleistet.
Seit einiger Zeit hat die Adventure-Familie Zuwachs bekommen. Ein neues Spielprinzip hat sich auf den CPC's mit "Shadowfire" vorgestellt. Von den Textabenteuern wurde ein ausgeprägtes Puzzlemoment übernommen, und die Arcaden-Vertreter der Gattung legten den Joystick mit in die Wiege. Der Name, den diese neue Unterabteilung der Abenteuer schlußendlich erhalten wird, steht noch nicht fest, aber sicherlich wird es etwas in der Art "Icon driven Adventure" oder "Menue-Adventure" sein. Ganz einfach, weil diese Namen mit treffender Sicherheit das Neuartige dieses Spielprinzips umschreiben. Wie bekannt, werden bei einem Textabenteuer die gewünschten Aktionen mittels mehr oder minder kom-



"Lords of Midnight" ist ein kombiniertes Strategie- und Abenteuer-Spiel. Es ist möglich, das Ziel des Spieles durch Schlachten mit rekrutierten Soldaten oder auf einem Abenteuerweg durch die Suche eines Einzelnen nach einem bestimmten Gegenstand zu erreichen.





Mit dem "Herrn der Ringe" liegt bereits das zweite Abenteuer von Melbourne House vor, in dem interaktive Charaktere mitspielen. Und sie spielen im wahrsten Sinne des Wortes mit – sind sie doch in der Lage, gewissermaßen einen eigenen Willen, der natürlich nur durch den Rechner simuliert wird, zu entwickeln.

plexer Eingaben erreicht, bei einem Arcaden-Abenteuer lenkt der Spieler in der Regel eine Spielfigur durch ein auf dem Monitor dargestelltes Szenario, das aus den verschiedensten Blickrichtungen gezeigt wird. Dieser neue Typ Abenteuer hat von beiden etwas übernommen. Alle den Spielfiguren möglichen Aktionen werden hier durch Menues aufgezeigt, und mittels eines Cursors, der sich frei in den Menues bewegen läßt, angewählt. Um beim Beispiel Shadowfire zu bleiben, versuche ich Ihnen das ganze Prinzip an diesem wohl ersten Vertreter der Menue-Abenteuer zu erklären. Es geht in dem Spiel darum, eine Spezial-Einsatztruppe, das Enigma Team, an Bord des Schlachtschiffes eines finsteren Imperators zu geleiten und einen dort gefangengehaltenen Botschafter zu befreien. Für jede Person des Teams sind im Spiel eine Reihe von Menues verankert, mittels derer sich die Figur bewegen beziehungsweise steuern läßt: für Handlungen, Bewegungen, Aktionen mit gefundenen oder mitgeführten Objekten und die allgemeinen Umstände je eines. Durch diese Menues kann der Spieler nun die jeweils von ihm gewünschte Aktion einleiten, indem er einfach den dafür vorgesehenen Punkt mit dem Cursor ansteuert und durch Drücken der Feuertaste an den Rechner weitergibt.

Die normalerweise in Adventures üblichen Messages, die Auskunft über Erfolg oder Mißerfolg der letzten Aktion geben, mußten hier einem großen Statusfenster weichen, in dem sich wiederum einige verschiedene andere Anzeigen befinden. Auf ihnen läßt sich ablesen, was sich zur Zeit gerade im Blickfeld der jeweiligen Person befindet. Sollte beispielsweise einmal der Schloßexperte beauftragt werden, eine verriegelte Tür zu öffnen, kann der Erfolg dieser Aktion in einem dieser Fenster, im vorliegenden Falle wäre es ein kleines Rechteck, durch das immer wieder die neuesten Meldungen gescrollt werden, abgelesen werden.
Das neueste Softwareprodukt, das dieser Art Abenteuer zuzurechnen wäre, ist "Mission Omega", das sich vom grundsätzlichen Spielprinzip doch um einiges von Shadowfire, unterscheidet. Hier geht es darum, in einer außerirdischen Raumstation einige versteckte Teile aufzufinden. Zu Beginn des Spieles muß der Spieler sich für diesen Zweck einige Androiden zusammenbasteln. Aus den verschiedensten Auswahlmöglichkeiten gilt es, die beste Fortbewegungsart, die beste Bewaffnung, die optimale Ausrüstung, und die am besten geeignete Energieversorgung zu finden.

Wie schon bei Shadowfire, unterliegt das Spiel auch hier einem Zeitlimit, nach dessen Ablauf das Spiel automatisch beendet ist und der Spieler, sofern er nicht noch in letzter Minute alle gestellten Aufgaben erfüllen konnte, das Spiel verloren hat.
Während des eigentlichen Spieles steht dem Spieler die Möglichkeit zur Verfügung, einen oder alle seiner Androiden auf Automatik zu schalten. In diesem Modus übernimmt der Rechner die Steuerung des Roboters. Eine weitere interessante Neuerung ist die Möglichkeit, den Androiden einen Weg vorzuprogrammieren, den diese dann ohne weiteres Zutun ganz von alleine zurücklegen, die einzige Voraussetzung hierfür ist, daß der Robot schon einmal durch die betreffenden Räume gelaufen sein muß.
Die vielen neuen Möglichkeiten, mit den Features zu experimentieren, machen das Spielen von Mission Omega außerordentlich interessant. Das Vorhandensein von abschaltbaren Energiegattern, die mancherorts ein Weiterkommen verhindern, würzen das Spiel noch zusätzlich, denn meistens ist es nicht gerade der Schalter nebenan, der das Energiegatter, das das weitere Vorankommen verhindert, abschaltet.
Im Gegensatz zu den althergebrachten Abenteuerspielen fließt der Nachschub hier noch relativ zögernd, sicherlich nicht zuletzt durch den immensen Programmieraufwand, der getrieben werden muß, um ein solches Menue-Adventure auf die Beine zu stellen. Aber das, was bisher vorliegt, macht einen wäßrigen Mund. Harren wir also der Dinge die da kommen.

Adventure International

Adventure International ist ein englisches Softwarehaus. Bei der Nennung dieses Namens kommen in mir die zwiespältigsten Gefühle auf.
Zum einen haben sich die englischen Programmierer, die die europäische Zweigstelle des US-Unternehmens repräsentieren, zumindest auf deutschem Boden sehr unbeliebt gemacht. Ganz gezielt ist dieser Imageverlust auf die deutsche Version des nach dem gleichnamigen Film erstellten Abenteuers "Gremlins" zurückzuführen. Diese Umsetzung kann ungelogen als gelungenes Beispiel dafür gelten, wie man es am be-

sten nicht machen sollte. Die vorgenommene Eindeutschung sieht so aus, als hätten die Engländer einen Liverpooler Hauptschüler, ausgestattet mit einem Deutschwörterbuch aus den 50-er Jahren, an ihren Quelltext gesetzt, und ihn von ihm unter Zeitdruck ins Deutsche übertragen lassen. Aber zu der unmöglichen Grammatik und der stellenweise verheerenden Satzstellung gesellen sich auch noch einige gravierende Fehler in der mitgelieferten Anleitung. Die dort aufgezählten Kommandos sind stellenweise schlicht und ergreifend falsch.
Im Grunde genommen ist Gremlins, solange man hier nur von der englischen Version redet, ein gutes Adventure-Spiel, aus dem man auch in Deutsch eine ganze Menge hätte machen können. Aber die Intention der Engländer, die sicherlich nicht billige Möglichkeit einen guten Übersetzer an die Sache zu setzen, aus Kostengründen unter den Tisch fallen zu lassen, hat sich im Endeffekt als die teurere Möglichkeit erwiesen. Denn um diesen Bock wieder auszubügeln, werden sich die Jungs etwas sehr Gutes einfallen lassen müssen.
Allerdings sollte man nicht zu hart urteilen, denn Adventure International war seit seinem Bestehen für seine guten Ideen bekannt. Zu Zeiten, als die Firma noch aus Scott Adams und seiner Frau bestand, heimste man allerorten die überschwenglichsten Kritiken für die phantasievolle Gestaltung der Produkte ein.
Wenn wir schon dabei sind von Scott Adams zu sprechen, will ich Ihnen aber auch nicht verschweigen, daß er es war, der erstmals ein Abenteuerspiel auf einem kleinen Heimcomputer realisierte. Bis zu diesem Zeitpunkt waren die Freuden des Abenteuerns einigen wenigen Personen vorbehalten, die Zugang zu großen Mainframe Rechnern hatten. Bis dahin war das Abenteuern auf Computern eher ein intellektuelles Vergnügen, mit dem sich Studenten auf ihren Institutsrechnern die Zeit vertrieben.
Nachdem Adams einige seiner Programme für die verschiedensten Rechnertypen realisiert hatte, mußte eine ständige Vertretung in England her, machte er dort doch mit seinen Programmen einen nicht unbeträchtlichen Umsatz. Adventure International GB war geboren.
Mit einer reinen Vertreibertätigkeit wollten sich die Engländer aber nicht zufriedengeben. Sie begannen an eigenen Projekten zu tüfteln und erzielten damit auch mehr als nur Achtungserfolge. Bald wanderten reine englische Programme zum Vertrieb nach Amerika. Ein gutes Beispiel für eine solche britische Produktion ist "Robin of Sherwood", das dann auch bezeichnenderweise einer englischen Fernsehserie nachempfunden wurde. Entgegen der üblichen Interpretation des Stoffes wurde hier ein Touch Fantasy mit in die Handlung integriert, der die nicht gerade neue Story so gelungen aufpeppt, daß sie sicherlich einem überdurchschnittlichen Anspruch gerecht wird.
Aber auch in punkto neuer Ideen und Spielkonzepte war man nicht gerade untätig. Der Anstoß zu diesen Neuerungen kam hier aus Großbritannien. Dort gibt es seit einiger Zeit eine Buchreihe, durch die man in die Lage versetzt wird, Abenteuer auch ohne den Computer zu erleben. Jedes dieser Bücher ist für sich ein spielbares Abenteuer, das nach folgendem Prinzip funktioniert:
Der Text ist in viele kleine Absätze unterteilt, die, jeder für sich, mit fortlaufenden Nummern versehen wurden. Lesen Sie einen solchen Abschnitt durch, wird Ihnen darin, wie auch auf den Computern, die Szene beschrieben, in der Sie sich gerade befinden. Stehen Sie beispielsweise gerade an einer Gabelung eines unterirdischen Tunnels, so haben Sie am Ende der Beschreibung die Wahlmöglichkeit, in der Sie sich entscheiden können, in welchem Tunnel Sie nun Ihren Weg fortsetzen wollen. Entscheiden Sie sich für den linken, lesen Sie bei Absatz Nummer 99 weiter, wollen Sie den rechten benutzen, lesen Sie bei 321 weiter.
Dieses neue Spielprinzip schlug wie eine Bombe ein. Dadurch, daß Kämpfe mit Untieren, die einem über den Weg laufen, nach Regeln ausgewürfelt werden, die dem Rollenspiel entlehnt wurden, stürzten sich die Fans dieser Spielegattung wie die Hyänen auf dieses neue Spielerlebnis. Die Fighting Fantasy Books wurden ein echter Renner. Die beiden Herren, die für dies alles verantwortlich zu machen sind, waren Steven Jackson und Ian Livingston. Zwar hatte Ian Livingston schon bei der Arbeit zu den vielgelobten Eureka-Aventures, die es leider nicht für die CPC's gibt, mit den Möglichkeiten der computerisierten Abenteuerei Tuchfühlung genommen, aber das unschlagbare Flair seiner Fantasy Bücher erreichte er hier, war er doch "nur" für die Story verantwortlich. Erst bei dem Adventure International Projekt "Seas of Blood", das sich inhaltlich ganz an das gleichnamige Fighting Fantasy Buch anlehnt, gelang es ihm, diesmal in Zusammenarbeit mit Steve Jackson, zu seiner alten Klasse zurückzufinden.
Es handelt sich hier im großen und ganzen um ein klassisches Abenteuer-Spiel, das aber durch die Übernahme der Kampfroutinen, wie sie auch im Buch stattfinden, einiges an Attraktivität und Spielbarkeit gewinnen.
Übrigens, wer sich für die Fighting Fantasy Books interessiert, dem sei geraten, sich an den Thieneman Verlag zu wenden. Dort ist die deutsche Übersetzung des ersten Spielbuches erschienen, es heißt "Der Hexenmeister vom Flammenden Berg" (Sorcerer of Firetop Mountain).

LEVEL 9

Eine englische Company, die sich bei der Pflege guter Textabenteuer besonders hervorgetan hat, ist Level 9. Zwar sind die Abenteuer dieser Firma auch mit Grafik versehen, allerdings ist diese sehr einfach gestaltet. Trotzdem ist es manchmal erstaunlich, wieviel die Programmierer in den Speicher eines kleinen Homecomputers hineinbekommen.
Schaut man sich beispielsweise "Red Moon" an, da sind Irrsinnsmengen Text, zusammen mit über 200 verschiedenen Bildern in den Speicher gepfercht worden. Aber wie erwähnt, sind die Grafiken nicht besonders ausgefeilt und werden mit einer kleinen Routine, die das fertige Bild aus einigen wenigen Datazeilen entstehen läßt, auf den Monitor gezaubert. Um einiges erwähnenswerter ist eigentlich der Text, den die Level 9 Jungs zu ihren Storys erstellen. Auch die reinen Storylines sind mit viel Geschick und Gefühl für Atmosphäre und Spannung erstellt worden.
Eines der interessantesten Level 9-Projekte ist eine dreiteilige SF-Saga, die in einer Raumstation beginnt, den Spieler im zweiten Teil auf die unwirtliche Oberfläche einer Dschungelwelt versetzt, in der es zu überleben gilt. Der dritte Teil handelt von der Suche nach einer legendären Androidenstadt, die im Verborgenen auf dem Dschungelplaneten existiert. Die Handlung dieser drei Abenteuerspiele macht deutlich, daß die Herren Programmierer ihre Lektion in Sachen anspruchsvoller SF gut gelernt haben. Die Titel hingegen vermitteln beim Betrachten im Vorübergehen nicht sehr viel von der Qualität, die sich hinter ihnen verbirgt. "Snowball", "Return to Eden" und "Worm in Para-

dise" sind auf jeden Fall Abenteuerspiele, die man sich zulegen sollte. Man wird zwar sicherlich einige Zeit brauchen, um sie zu lösen aber nichtsdemzutrotz sind sie beste Abenteuerunterhaltung.

Das neueste Level 9-Projekt wendet sich thematisch wieder der Fantasy zu. "The Prize of Magick" handelt von einem magischen Zeitalter, in dem es immer schwieriger wird, Zauberei einzusetzen. Ein junger Adept macht sich daraufhin auf den Weg, um herauszufinden, woran dieses Verschwinden der Magie liegt.

Wie gewohnt, sind auch dieses Mal über 200 verschiedene Lokationen, die jeweils mit einem Bild versehen worden sind, in das Spiel integriert worden. Die Texte weisen die gewohnte Level 9-Qualität auf. Einen kleinen Nachteil haben aber alle Level 9-Spiele gemeinsam, aufgrund der Tatsache, daß in England eine Floppy-Disc eher die Ausnahme, denn die Regel ist, sind die Programme in ihrer gesamten Struktur auf den Kassettenbenutzer ausgelegt. In Anbetracht der teilweise immensen Ladezeiten wurde eine mehrteilige Konzeption, in der einzelne Programmteile aus dem Speicher gelöscht werden und durch neu geladene Files ersetzt werden, von vorneherein ausgeklammert. Die Programme laden in einem Stück und verbleiben so wie sie geladen wurden im Speicher des Rechners. In Anbetracht der Möglichkeiten, die sich durch den geschickten Einsatz einer Floppy bieten, muß man feststellen, daß dort ein wenig der Anschluß verpaßt wird. Herrje..., was könnten die Jungs doch für tolle Abenteuer auf die Beine stellen – das Zeug dazu haben sie!



Activision

Auf dem Abenteuersektor ist Activision im Grunde genommen nicht besonders aktiv. Zwar haben sie das eine oder andere Spiel im Sortiment, für die CPC-Benutzer ist davon aber nur eines interessant: "Mindshadow", das sowohl von seiner Grafik her wie auch von der Story, die dort geboten wird, eines der besten Abenteuer des Jahres 1986 darstellt.

Das Spiel beginnt auf einer einsamen Südseeinsel, auf der eine kleine Hütte steht. In dieser Hütte erwacht die Spielfigur dann auch das erste Mal und muß feststellen, daß sie ihr Gedächtnis verloren hat. Ziel des Spieles ist es, das Gedächtnis wieder zu erlangen; aber dazu muß die Spielfigur erst einmal die Insel verlassen, und das erweist sich als schwieriger, als es auf den ersten Blick aussieht.

Auch Mindshadow zählt zu den grafikunterstützten Abenteuerspielen. Die Bilder, über die das Spiel verfügt, sind von, für CPC-Verhältnisse, allererster Qualität und werden in einem atemberaubenden Tempo auf dem Monitor aufgebaut.

Zum ersten Male ist in Mindshadow eine softwaremäßige Quicksave-Routine eingebaut worden. Mittels des einfachen Befehles 'Quicksave', hat der Spieler die Möglichkeit, den Spielstand abzuspeichern, ohne auf den Massenspeicher, egal welcher das nun sein möge, zuzugreifen. Mittels 'Quickload' ist der vorher abgespeicherte Spielstand ebenso schnell wieder in den Speicher zurückgeholt, wie er abgesaved wurde.

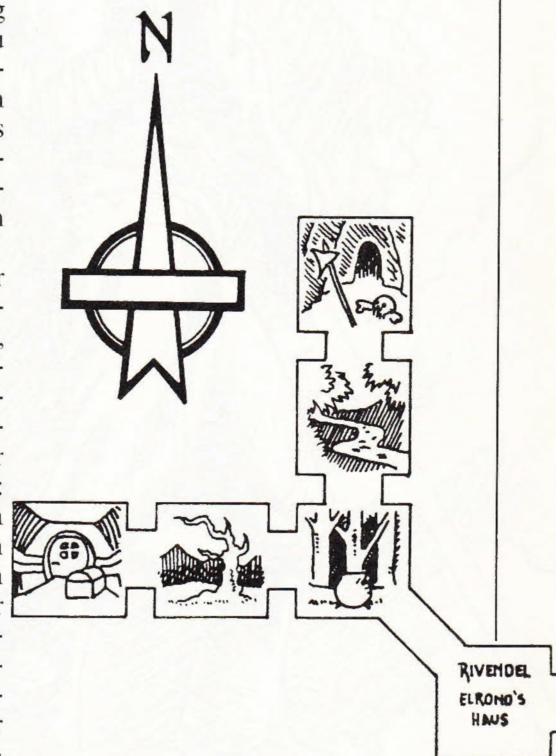
Schaltet man den Rechner allerdings ab, ohne vorher einmal einen Hardwaresave durchzuführen, ist der Spielstand natürlich verloren.

Eine weitere erwähnenswerte Tatsache ist das Tutorial, das sich mit auf der Kassette befindet. Hier wird der Adventure-Anfänger mit viel Einfühlungsvermögen in das doch ein wenig gewöhnungsbedürftige Spielprinzip der Abenteuer eingeführt.

Mindshadow ist ein Spiel, an dem sowohl Anfänger wie auch der Fortgeschrittene seine helle Freude haben wird. (HS)

DER HERR DER RINGE

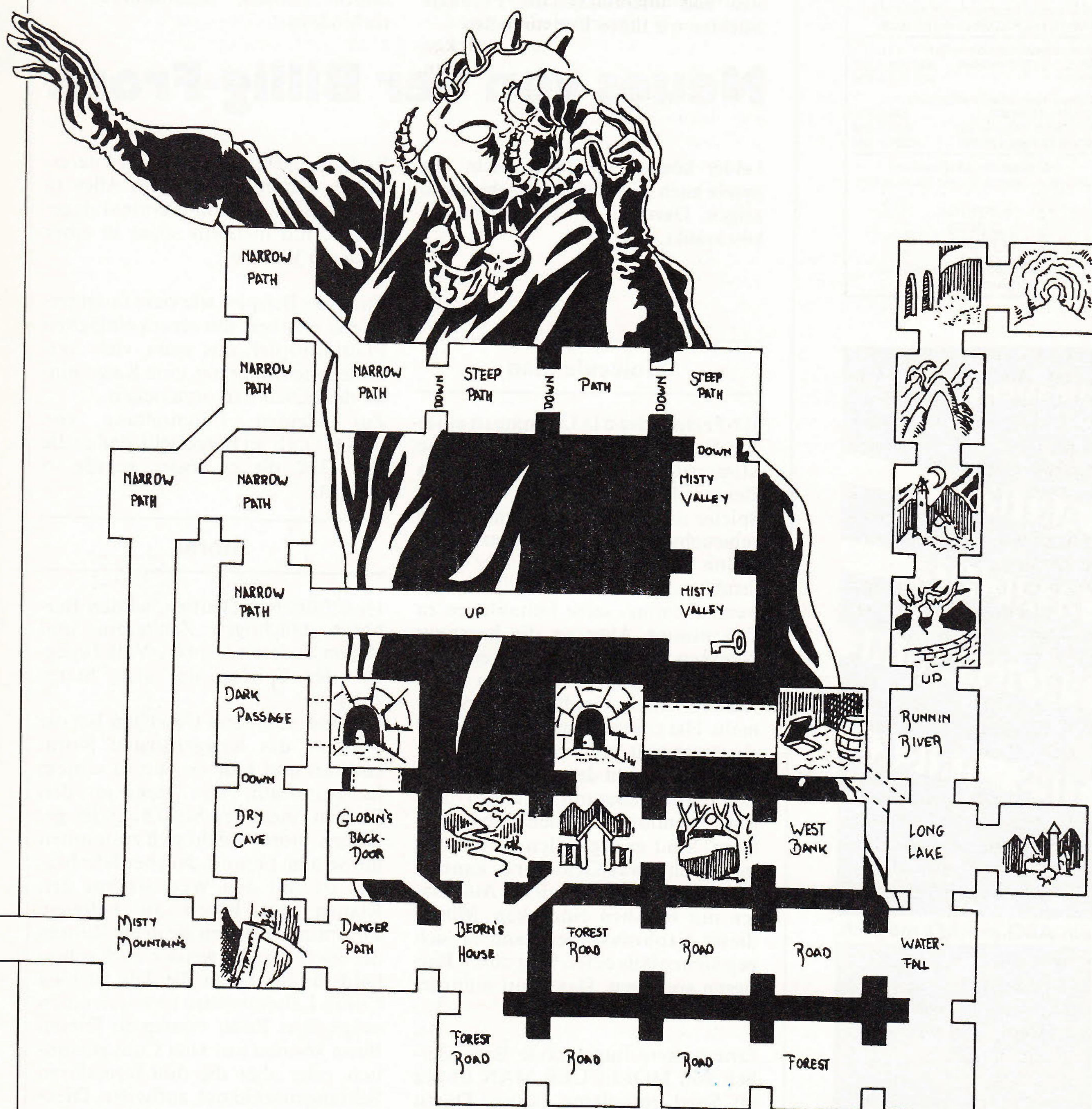
Das Programmierer-Team um Philip Mitchel hat wieder zugeschlagen. Bereits der erste Versuch dieser Crew, die phantastische Welt Professor Tolkiens als Computerabenteuer umzusetzen, kann als voller Erfolg bezeichnet werden. Nicht zu Unrecht wird der "Hobbit" als der Klassiker der Abenteuerspiele bezeichnet. Obwohl schon mehr als drei Jahre alt, gelang es ihm doch, sich bei der CPC International-Wahl der besten Software des Jahres in der Sparte Abenteuer für Platz Eins zu qualifizieren. Stellte die Geschichte, die in "Der kleine Hobbit" erzählt wird, die Vorgeschichte des Herrn der Ringe dar, konnte man sich eigentlich an zwei Händen abzählen, wie lange es nun dauern würde, bis jener seinen Weg in die Homecomputer finden würde.
Aber lange Verhandlungen um die Titelrechte verzögerten die Fertigstellung des Programmes ungemein. So dauerte es, nach der ersten Ankündigung, mehr als zwei Jahre, bis der Herr der Ringe für die Computerwelt zur Realität wurde.
Aber nun ist die Warterei vorbei – es ist soweit. Als das Paket von Melbourne House bei uns eintraf, dachten wir zuerst, man hätte uns außer der Kopie des "HDR" noch einige andere Software zum Testen zugesandt. Weit gefehlt, denn die Box, in der der "HDR" geliefert wurde, hat alleine eine Dicke von mehr als fünf Zentimeter. In dieser wirklich gediegenen Aufmachung ist dieses Programm ein echter Schmuck für jede Softwaresammlung. Nicht ohne Grund ist die Packung so dick, denn es ist, wie schon beim Hobbit, in der Verpackung ein Exemplar der literarischen Vorlage enthalten. Eingefleischte Abenteuer- und Fantasy-Fans werden nun stutzen und sagen: „Was! Ein Buch nur? Der HDR umfaßt aber doch insgesamt drei Bücher!" Richtig erkannt. Das vorliegende Programm stellt, wie das darin enthaltene Buch, auch nur den ersten Teil der Ringsaga dar: "Die Gefährten" bzw. "The Fellowship of the Ring" wie der Titel im englischen Original lautet. Die fehlenden beiden Teile befinden sich, soweit man Melbourne House dahingehend trauen kann, bereits in Arbeit.
Eine Schilderung der Inhalte einer derart episch breit angelegten Erzählung, wie sie der HDR ist, kann selbstverständlich nur Stückwerk bleiben. Trotzdem möchte ich versuchen, all denen, die sich in der Geschichte von Mittelerde nicht so gut auskennen, zumindest einen Eindruck von der Atmosphäre, die auf sie wartet, zu vermitteln.
Im Anfang der Zeiten schmiedeten die Elben die Magischen Ringe der Macht, und verteilten sie unter den Völkern von Mittelerde. Sie sollten all den Zwergen, Menschen, Hobbits und Elben Schutz vor den Mächten der Finsternis bieten. Im dunklen Lande Mordor überlegte Sauron, der Fürst der Finsternis, wie er sich die Macht der Ringe zu Eigen machen könnte.
Sauron täuschte die Elbenschmiede und es gelang ihm, von ihnen die Fertigkeit des Ringeschmiedens zu erlernen. Wieder nach Mordor zurückgekehrt, machte er sich sogleich daran, einen Ring zu schmieden. Und alles was dunkel und schlecht war, wurde in diesem einen, dem Meisterring, geschmiedet. Ausgerüstet mit dieser machtvollen Waffe machte sich Sauron nun daran, alle Länder von Mittelerde mit Finsternis zu überziehen. Jedoch gelang es einem der Elbenfürsten, Sauron im Zweikampf zu besiegen und den Meisterring von seiner Hand zu schlagen. Doch die Macht des Ringes war größer als man glaubte. Wie von einem eigenen Bewußtsein erfüllt, versuchte der Ring den Geist der Person, die ihn trug, zu verändern. Doch bei dem reinen Gewissen, das ein Elbe hat, gelang ihm das nicht. Als dieser Fürst allerdings auf der Flucht vor einer mörderischen Orkhorde einen Fluß überquerte, glitt ihm der Ring vom Finger.
Von nun an war der Ring für Jahrhunderte verloren. Erst ein kleiner Hobbit, nämlich Bilbo Beutlin, entdeckte ihn auf einer abenteuerlichen Schatzsuche tief unter den Felsmassiven der Nebelberge. Dort hauste ein Wesen, das ganz und gar der Macht des Ringes verfallen war: Gollum. Nachdem Bilbo ihm den Ring abgerungen hatte, begab sich auch Gollum auf die Suche nach dem verlorenen Ring. Im Laufe seiner Abenteuer, (die übrigens den Hintergrund des Hobbits darstellen) erlangte Bilbo auch den Schatz des letzten Drachens. Aber auch in Mordor fühlt Sauron, daß der Ring wieder gefunden war. In den Jahren des ersten Ringkrieges ist es ihm gelungen, all die anderen Ringe der Macht in seinen Besitz zu bringen, und deren Natur ins Schlechte zu verkehren. Nun stehen ihm die Ringgeister, die Nazgul, zu Gebote. Diese sendet er auf Dämonenrossen durch das Land, um den einen, den Meisterring, für ihren Herrn zurückzuerlangen. Aber auch Elben und Menschen haben die Gefahr, die ihnen von diesem Ring droht, erkannt und so beschließt man, eine kleine Truppe auszusenden, die versuchen soll, den Ring zu vernichten. Aber der einzige Ort, an dem dies gelingen könnte, ist auch der Ort, an dem der Ring geschmiedet wurde. In den Feuern des Schicksalsberges, einem Vulkan im Herzen des dunklen Landes Mordor.
Vor diesem farbigen Hintergrund findet also nun die Handlung des HDR statt. Ganz klar, daß sich eine solch umfangreiche Geschichte nur schwer in einem Programm unterbringen läßt. Zumal man auch noch Probleme mit den einzelnen Personen



bekommt. Immerhin sind es bereits zu Anfang drei Hobbits, die zu der gefahrvollen Reise aufbrechen. Aber auch daran haben die Programmierer von Melbourne House gedacht. So ist es möglich, während des Spiels die Figur zu bestimmen, die man gerade darstellen möchte. Man gibt ganz einfach "become Frodo" ein, und schon schlüpft man in die Haut der entsprechenden Person. Auf diese Art und Weise ist es möglich, mit mehreren Personen gleichzeitig zu spielen. Natürlich sind die Personen, die dem Spieler im Laufe des Abenteuers begegnen, auch größtenteils interaktiv. Das heißt, sie sind in der Lage, eigene Entscheidungen zu treffen. Beispielsweise kann es Ihnen passieren, daß Sie eine Person fragen, ob sie Sie vielleicht tragen könnte (Hobbits haben kurze Beine). Fragen Sie das erste Mal, lehnt diese Person ab, beim zweiten Mal ebenfalls. Nun gehen Sie einen Schritt zurück, kehren dann zur betreffenden Person zurück und fragen abermals, und plötzlich gewährt er Ihre Bitte. Ganz klar, daß ein solches Feature ungemein zur Atmosphäre eines solchen Spieles beiträgt.

Zwar hat auch der HDR Grafiken im Programmablauf eingebettet, jedoch sind diese so spärlich gesät und von der Machart so bescheiden (wahrscheinlich zu Gunsten eines komplexeren Programmes), daß man sie bestenfalls als leichte Unterstützung ansehen kann. Wie schon beim Hobbit ist es auch hier möglich, ganze Sätze als Kommandos einzugeben. Ein sehr umfangreiches Vokabular ist da schon fast selbstverständlich.

Resümiered möchte ich zum Hobbit noch sagen, daß sich das lange Warten auf jeden Fall gelohnt hat. Für die Adventures ist es sicherlich der Knaller der Saison. (HS)



SPIELEKARTEN

Neues von der Billig-Front

Die Budget Labels sind erwachsen geworden. Nachdem sie noch vor einem Jahr, ein wenig abfällig als Supermarkt-Wühltischsoftware bezeichnet, eher noch belächelt denn ernstgenommen wurden, haben sie heute eine nicht mehr wegzudenkende Position inne. Daß sich ihr Marktanteil seither noch weiter vergrößert hat, ist fast schon selbstverständlich.
Billigsoftware - das ist inzwischen schon fast so etwas wie ein Qualitätsbarometer geworden, anhand von Budget Spielen kann der Anwender ermessen, was sein Geld inzwischen Wert ist.
Wie bei den „Normal Prized" Produkten, hat das Jahr 1986 auch bei den Budget Labels eine wahre Softwareschwemme gebracht. Einige der neuesten und interessantesten Produkte möchten wir Ihnen hier vorstellen.

Leider können wir Ihnen nicht alle Spiele auch mit einem Bildschirmfoto zeigen. Dazu ist der Platz leider zu beschränkt.

Molecule man

3D-Perspektive á la Ultimate ist eines der Features, über die dieses neue Mastertronic Spiel verfügt. Einen kleinen Ball mit Beinen muß der Spieler durch ein großes, radioaktiv verseuchtes Labyrinth steuern. Der kleine Kugelkerl kann die harte Strahlung aber nur dann überleben, wenn er immer seine Jodtabletten zu sich nimmt. Aber an die heranzukommen ist gar nicht so einfach. Zuerst muß der Spieler die im Spielfeld verteilten Geldstücke einsammeln. Hat er erst einmal genug davon eingesammelt, gilt es einen Automaten zu finden, an dem der kleine Kerl sich dann wieder mit seinen so dringend benötigten Pillen eindecken kann. Sind genug Pillen eingekauft, und noch etwas Geld übrig, kann er sich noch an einem anderen Automaten mit Bomben eindecken. Mittels dieser Explosivkörper kann er sich gegebenenfalls einen Weg durch Barrieren sprengen. Das kann mitunter das Spielleben retten.

Eine weitere interessante Besonderheit von MOLECULE MAN ist der im Spiel enthaltene Editor. Damit kann der Spieler sich seine eigenen Szenarios zusammenbasteln und auf Kassette abspeichern.
MOLECULE MAN ist ein gutes Beispiel dafür, was man am richtigen Ort für 12,– DM geboten bekommt.

Radzone

Obgleich das Thema Radioaktivität seit Tschernobyl ein wirklich heißes Eisen ist, muß es dennoch immer wieder herhalten, um irgendwelchen kleinen Männchen das Leben schwer zu machen. So hat bei RADZONE ein einzelner Mann (natürlich der Spieler) die Aufgabe, eine Reihe verlorengegangener radioaktiver Objekte einzusammeln, um so unter Einsatz seines Lebens, Schlimmeres zu verhindern.

Radzone ist eine der weniger interessanten Neuerscheinungen. Alles in diesem Spiel ist schon einmal dagewesen, und meistens sogar in einer besseren Version.

Ein gutes Beispiel, wie viele Programmierer glauben, mit einem einfachen Plattformspiel, das extra viele verschiedene Bilder hat, eine Katze hinter dem Ofen hervorzulocken.
Zur kleinen Ehrenrettung von RADZONE sei festgestellt, daß es die 12,– DM, die es kostet, gerade so wert ist.

Storm

Unheimlichen Grüften, wilden Barbaren, mächtigen Zauberern, und vielem anderen Fantasy-Volk begegnet der Spieler, der sich Storm zulegt.
Der böse Zauberer Una Cum hat die Verlobte des Kriegerfürsten Storm entführt und hält sie nun in seinem Laboratorium, das sich in den Ruinen einer alten Stadt befindet, gefangen. Storm macht sich zusammen mit seinem Freund, der ebenfalls Magier ist, auf den Weg, sie aus den Klauen des Bösen zu befreien. Zusammen dringen sie in die Ruinen der Stadt ein, und werden auch schon bald fündig. Doch die Tür zu Una Cum's Laboratorium ist durch einen magischen Bann verriegelt. Diesen Bann können nur Una Cum persönlich, oder aber die drei legendären Schlangenschlüssel, aufheben. Diese drei Schlüssel sollen der Sage zufolge

auch in diesen Ruinen verborgen sein. Storm und Agravain, der Magier, beschließen, sich auf die Suche nach den Schlüsseln zu machen – ist das doch die einzige Möglichkeit, in Una Cums Sanktuarium einzudringen. Allerdings hat Una auch für diesen Fall Vorsorge getroffen und in jedem Haus der Ruinenstadt magische Generatoren aufgestellt. Sobald ein Fremder den betreffenden Raum betritt, beginnen die Generatoren Krieger auszuspucken, zwar nur Scheinbilder, aber Scheinbilder die töten können.

Der einzige echte Mangel der Storm innewohnt, ist die Tatsache, daß es sich ausschließlich mit der Tastatur spielen läßt. Aber wie gesagt, das ist der einzige Mangel. Ansonsten ist Storm ein richtig gutes Arcaden-Abenteuer, das sogar ein Feature hat, das man nicht überall findet. Es ist möglich, in einem Spiel gleichzeitig mit zwei Spielern zu spielen. So lassen sich die zu bewältigenden Aufgaben noch ein wenig leichter meistern.

Die Grafik präsentiert sich in einer akkuraten, bunt gestalteten Aufmachung. Prädikat: Empfehlenswert.

Spiky Harold

Harold, der Igel, macht sich für den Winter bereit. Für den Winterschlaf braucht er eine anständige Fettschicht unter seinen Nadeln, will er den folgenden Frühling auch erleben.
Anfangs macht sich Harold keine großen Gedanken, allerdings, je weiter er auf seiner Suche vorankommt, desto schwieriger wird das Einsammeln seiner Leckereien. Aber Harold ist guten Mutes, bisher hat er es noch jeden Herbst geschafft, genug Fett auf die Rippen zu bekommen, um damit den Winter wohlbehalten zu überstehen.

Bei SPIKY HAROLD von Firebird handelt es sich, ebenso wie bei Radzone, um ein Plattformspiel. Nur gibt es einen großen Unterschied. SPIKY HAROLD ist um einiges besser.

Die Grafik ist zwar nicht so farbenfroh wie bei Radzone, aber trotzdem um einiges besser gelungen als dort. Der Spieler muß den gefräßigen Igel durch 59 verschiedene unterirdische Räume lenken. In jedem dieser Räume findet Harold etwas zu fressen, oder Goldmünzen. Sammelt er diese ein, bringt das ein Extra-Leben. Auch hat dort jemand irgendwann eine ganze Menge Schnaps verloren. Findet Harold eine solche Flasche und frißt sie, wird er daraufhin auch postwendend betrunken.

SPIKY HAROLD ist vielleicht vom Schwierigkeitsgrad ein wenig hoch angesetzt, aber sonst ein Spiel, das man durchaus als gut bezeichnen kann.

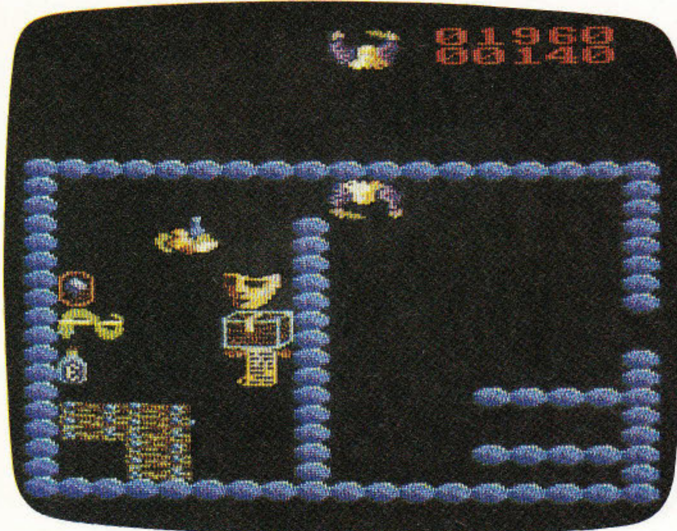
Thrust

Bevor ich Ihnen überhaupt etwas über THRUST erzähle, ein sehr guter Tip: Kaufen Sie sich THRUST so schnell wie möglich, bevor jemand auf die Idee kommt, den Preis zu erhöhen. Hier liegt nämlich ein Spiel vor, das, selbst wenn es 30,– DM kosten würde, noch jeden Pfennig doppelt wert wäre. Im Grunde genommen geht es lediglich um das Abschleppen von ganz bestimmten Gegenständen von unterschiedlichen Planeten. Die Steuerung des Raumschiffes ist denkbar einfach, links und rechts rotieren, Gas geben, tanken bzw. docken. Das sollte der Spieler schnell lernen, denn schon auf dem ersten Planeten wird klar, daß es gar nicht so einfach ist ‚herumzufliegen, wenn man sich dabei permanent mit der Schwerkraft des Planeten auseinandersetzen muß.

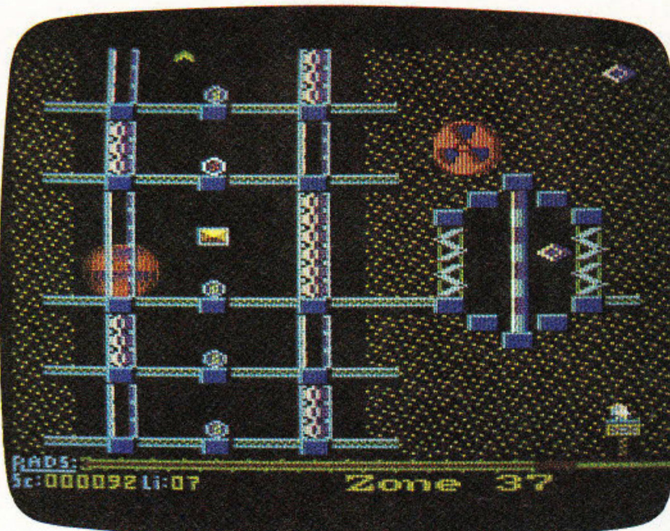
Aber nicht nur die Anziehungskraft macht dem Piloten das Leben schwer. Immer wieder begegnet er Geschützstellungen, die einem weiteren Vordringen im Wege stehen. Der einzige sichere Weg, sich Ärger mit diesen Geschützen zu ersparen, ist, sie aus sicherer Entfernung abzuschießen. Nur ist das nicht immer möglich, verlegt sich der Schauplatz dieses Schwerkraftdramas bereits im zweiten Level unter die Erde. Durch die Gravitation ist das Manövrieren des Raumschiffes nicht gerade leicht, hat man allerdings erst einmal den Pod gefunden und mit ihm gedockt, fangen die Probleme erst richtig an. Der Pod hängt nämlich nun an einer langen Trägertrosse unter dem Raumschiff, und schwankt dabei auch, je nachdem, wie Sie fliegen, mehr oder weniger hin und her.

THRUST, was soviel wie Beschleunigung bedeutet, macht ohne zu übertreiben süchtig. Je weiter der Spieler in die unterirdischen Kavernen vordringt, desto schwerer, aber auch interessanter wird es. Ein Spiel, das eine derartig hohe Spielmotivation erzeugt, habe ich schon lange nicht mehr gesehen.

Ich kann nur wiederholen was ich zu Beginn dieser Review sagte: Holen Sie sich dieses Spiel, Sie werden es sicherlich nicht bereuen! (HS)



STORM



RADZONE

Das Traumtool

Adventures erfreuen sich einer allseits wachsenden Beliebtheit. Sei dies nun bei jenen Leuten, die sich gerne vor ein gut gemachtes gekauftes Computerabenteuer setzen, oder bei denen, die sich von der programmiertechnischen Seite her nähern, und sich ihre Adventures selber schreiben. Nun gibt es ein Programmiertool, das sich an die Vertreter beider Seiten wendet.

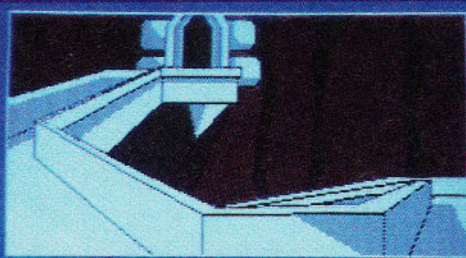
Der Graphic-Adventure-Creator

Sicherlich haben viele unserer Leser schon einmal eine schlaflose Nacht vor ihrem Computer verbracht, nur weil sie an dem vermaledeiten Höhlenschrat nicht vorbeikommen. Sei das Problem, das sich ihnen in den Weg stellt, nun ein Höhlenschrat, ein Drache oder eine verschlossene Tür, eines ist ihnen allen gemeinsam, sie sind der Bestandteil eines Adventures. Nicht minder viele Leute haben aus einem anderen Grund ihren Rechner nächtelang nicht ausgeschaltet: Sie brüten vor einem Programm, das sie gerade erstellen. Fest steht außerdem noch, daß sich ein guter Teil der Programmierer, die sich die Nächte um die Ohren hauen, an Adventures herumhacken. In dem hier anstehenden Test wollen wir Ihnen ein Programmiertool vorstellen, das gewiß viele Adventurespieler und noch mehr Adventureschreiber interessieren wird. Nämlich ein Programm, mit dem sich auf einfachste und sehr zeitsparende Art und Weise die spannendsten Adventures erzeugen lassen. Lassen Sie sich aber nun von dem Wort "erzeugen" nicht erschrecken, denn natürlich erzeugt der Graphic-Adventure-Creator nur das endgültige Programm aus den von Ihnen vorgegebenen Fakten. Also ist das erste wirklich interessante Feature, über das der GAC verfügt, die Tatsache, daß alle mit ihm erstellten Spiele komplett alleine laufen; also einen Code erzeugen, der "Stand allone" auf Ihrem Rechner läuft. Lassen Sie doch erst einmal folgende Fakten auf sich wirken: bis zu 9999 verschiedene Räume, 255 Objekte, 255 Meldungen, 255 Verben, 255 Substantive, 255 Adverben. Also wenn sich aus diesen Zutaten kein gutes Abenteuer brauen läßt, dann mit gar nichts. Doch damit sind noch nicht alle Features über, die der GAC verfügt, aufgezählt. Auch für die, die der Meinung sind, daß Graphikadventures das Allergrößte seien, ist der GAC das optimale Werkzeug.

Wie der Name schon verrät, ist in dem Programm ein Graphiktool mit eingearbeitet. Und auch die mit diesem Tool erstellten Bilder lassen sich in das Programm miteinbinden. Allerdings müssen wir hier auch darauf hinweisen, daß ohne eine zündende Idee selbst so ein ausgefeiltes Werkzeug wie der GAC nutzlos ist. Denn am Anfang eines guten Abenteuers steht immer eine gute Story. Sie sollten sich also zuerst einmal überlegen, um was es eigentlich in Ihrem Abenteuer gehen soll. Man braucht dabei eigentlich gar nicht so stur sein und sich diese Geschichte selbst ausdenken – manchem fällt eine solche Aufgabe sowieso nicht sehr leicht, überlegen Sie doch einmal ein bißchen. Sicherlich ist Ihnen in letzter Zeit ein Buch besonders in Erinnerung geblieben; vielleicht läßt sich ja dessen Story mit leichten Änderungen adaptieren. Mit ein wenig Phantasie kann man sogar aus einem Märchen von den Gebrüdern Grimm ein fetziges Abenteuer machen. Haben Sie nun eine geeignete Geschichte? O.K., lassen Sie uns sehen, wie es weitergeht.

Sie müssen sich nun eine Karte zeichnen, in der Sie Ihre Handlung unterbringen. Am besten gehen Sie bei dieser Arbeit mit dem Papier nicht zu sparsam um. Liegt die großflächig gezeichnete Karte nun vor Ihnen, so schreiben Sie nun in die einzelnen Räume jeweils das, was wichtig in und an ihnen ist. Z.B.: Wie heißt der Raum, welche Objekte befinden sich in ihm, soll etwas Besonderes passieren, wenn der Spieler diesen Raum betritt, oder soll etwas Besonderes durch bestimmte Handlun-



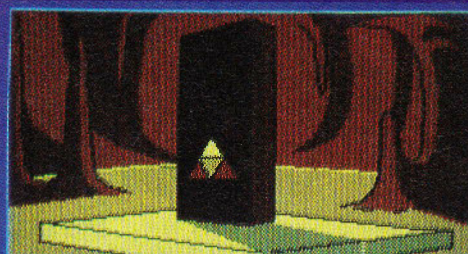


Das Traumtool

gen ausgelößt werden. Außerdem vermerken Sie dann noch, welche der Objekte der Spieler aus dem Raum entfernen kann, und welche stationär sind. Wenn Sie sich jetzt noch überlegen, welche Dialoge bzw. Texteingaben und Antworten, die in jedem Raum stattfinden, benötigt werden und sich diese auch noch, falls Sie dazu noch Platz haben, auf Ihrer Karte notieren, sollten Sie eigentlich alles Material zusammen haben, um ein gutes Abenteuer zu schreiben. Für das weitere Vorgehen gibt es nun verschiedene Möglichkeiten: Entweder Sie fangen nun an und generieren das Programm, oder aber Sie erstellen die für das Spiel benötigten Illustrationen. Würden wir vor dieser Wahl stehen, so würden wir uns für die Bilder entscheiden. Hier zeigt sich der GAC auch gleich von seiner besten Seite.
Der Grafikeditor ist nämlich sehr ausgefeilt, und bietet eine Anzahl Optionen, die man zwar von normalen Grafikprogrammen her kennt, die hier aber durch einen Trick ein wenig vervollkommnet wurden. Der GAC merkt sich jeden vom Anwender gesetzten Punkt und legt ihn in einer Tabelle ab. Durch diese Tabelle kann man blättern, und irrtümlich gesetzte Punkte, gezogene Striche oder gemalte Kreise, Rechtecke, ausgefüllte Flächen oder was man sonst noch in seiner Grafik hat, läßt sich genauso gut, wie es auf den Bildschirm gebracht wurde, auch wieder entfernen. Einzige Option, die wirklich vermißt wurde, war eine Lupe. Bilder lassen sich zusammenfassen und übereinander kopieren. Zum Erstellen einer Grafik stehen dem Programmierer vier Grundfarben pro Bild zur Verfügung, die untereinander noch schattiert werden können. Daß sich damit eine ganze Menge anfangen läßt, beweist das Demoadventure RANSOM, welches sich auf der GAC-Diskette befindet, auf recht eindrucksvolle Weise.
Nach dem Erstellen der bis zu 999 Bilder, legt man in einer Verbendatei alle im Programm erlaubten Tätigkeitswörter fest. Dabei ist es völlig egal, ob Sie nun deutsche Texte oder irgendeine andere Sprache verwenden. Bei Persisch und Kyrillisch treten wegen des Zeichensatzes Schwierigkeiten auf, aber da es sehr wenig Adventure in einer dieser Sprachen gibt, kann das nicht als Fehler gerechnet werden.
Nach den Verben verfährt man genauso mit Objekten, Beschreibungen von Objekten, Beschreibungen von Lokationen und Antworttexten. Sind die Wortdateien erstellt, geht es zum ersten Mal an Prioritäten von Bedingungen. Dies ist zwar immer noch sehr einfach zu handhaben, aber spätestens hier muß man sich darüber im klaren sein, mit welchem Schlüssel, welche Tür geöffnet werden kann, und in welchem Raum dieser Schlüssel zu finden ist. Die Eingabe ist, wenn man sich die Struktur festgelegt hat, durch eine Tabellenform kinderleicht und Verknüpfungen sind sofort klar zu durchschauen.
Der Hauptpart, die eigentliche Adventurelogik, macht die wenigste Arbeit. Um die Bedingungen für Ausgänge, Räume und Bilder festzulegen, benötigt man nur minimale Kenntnis der englischen Sprache. Diese Logik wird in Zeilen von Befehlen geschrieben, und enthält ein Kriterium, auf das der Rechner eine bestimmte Aktion ausführen soll. Die Aufsplittung des eingegebenen Satzes erledigt der Parser automatisch.
Alle Daten über das Adventure können selbstverständlich zwischengespeichert werden und somit jederzeit auf Drucker zu Papier gebracht werden.
Haben Sie alle Angaben gemacht, die Grafiken gemalt und alle Logikfehler korrigiert, so können Sie einen Testlauf starten und nach erfolgreichem Abschluß Ihr eigenständiges Adventure kreieren lassen.
Das umfangreiche deutsche Handbuch ist mit Referenzkarten und Beispielen versehen, so daß ein erfolgreiches Arbeiten mit GAC überhaupt kein Problem sein sollte.
Auf der zu diesem Heft erhältlichen Databox ist ein Demoprogramm des GAC enthalten.

(TM/HS)





CPC DATABOX

— Das ist die Software zum CPC Magazin —

DATABOX:

— mehr als der übliche Softwareservice

— bringt alle sechzehn Programme des Sonderheftes auf Kassette oder 3''-Diskette

— Die Programme sind, durch den auf den Datenträger enthaltenen CPC-Emulator, auf allen drei CPC-Rechnern lauffähig.

— trägt das Titelbild des gleichzeitig erscheinenden Sonderheftes.



Für alle CPC's als Kassette und 3'' Diskette.

Inhalt der Databox zu Sonderheft 3/86

Galactic Conflict
Hubschrauber Simulator
Gangster
Elite-Commander
Mühle
Wipo
Monstergarten
Checksummer
CPC-Emulator
Entengeschichten
The Designer
Music Interrupt
CPC-Monitor
Listenverarbeitung
Tape Monitor
Disk-Patcher
Tabwrite





Einzelbezugspreise für DATABOX:
Diskette 3'' 29,– DM zuzüglich 3,– DM Porto/Verpackung (im Ausland zuzüglich 5,– DM Porto/Verpackung).

Kassette 19,– DM zuzüglich 3,– DMPorto/Verpackung (im Ausland zuzüglich 5,– DM Porto/Verpackung).

Bitte benutzen Sie den Bestellcoupon!

Name: ....

Anschrift: ....

Hiermit bestelle ich die Databox zum Sonderheft 3/86
◇ als Diskette zum Preis von 29,– DM
◇ als Kassette zum Preis von 19,– DM
zuzüglich 3,– DM Porto/Verpackung
im Ausland zuzüglich 5,– DM Porto/Verpackung
◇ bitte liefern Sie per NN
◇ V-Scheck liegt bei

...hneider CPC International
Postfach 250, 3440 Eschwege

Knight Lore

Auch Knight Lore ist ein fast schon klassisches kombiniertes Labyrinth und Jump & Run-Spiel, das noch aus den Zeiten stammt, als "Ultimate" allerorten als Vorreiter der Branche mit immer neuen Ideen galt. Viele unserer Leser haben uns zu diesem Spiel Karten zugeschickt. Dafür möchten wir uns auch herzlichst bedanken.
Ganz genau hat Ute Pogrzeba das Labyrinth des Zauberers erkundet. Die Gegenstandsliste hat sie erarbeitet. Die Karte ist eine Coproduktion unserer Grafiker und all den Knight Lore-Kartographen, die uns ihre Erfahrungen mitgeteilt haben.
Bei jedem neuen Spielbeginn gibt es acht Möglichkeiten (siehe Liste), in welcher Reihenfolge die Gegenstände in den Räumen vorzufinden sind. Noch ein Tip: Den Raum des Zauberers, in dem man die gewünschten Gegenstände in den Topf wirft, sollte man nur bei Tag betreten.

1. Möglichkeit:
Giftflasche (1)
Diamant (2)
Schuh (3)
Tasse (4)
Männchen (5)
Flasche (6)
Kelch (7)
Kugel (8)

2. Möglichkeit:
Diamant (1)
Männchen (2)
Giftflasche (3)
Kelch (4)
Kugel (5)
Tasse (6)
Schuh (7)
Flasche (8)

3. Möglichkeit:
Kugel (1)
Flasche (2)
Männchen (3)
Giftflasche (4)
Tasse (5)
Schuh (6)
Diamant (7)
Kelch (8)

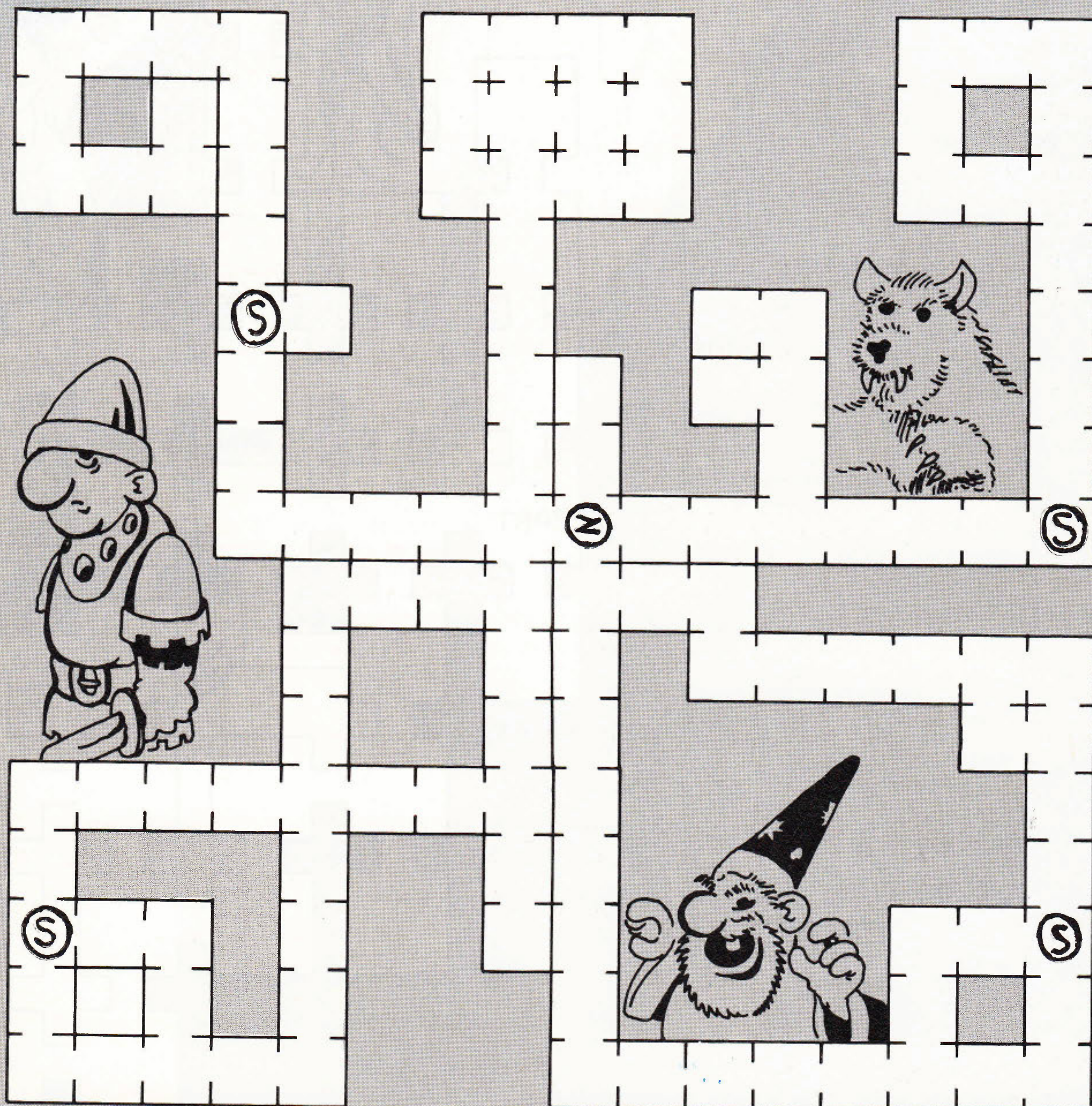
4. Möglichkeit:
Kelch (1)
Schuh (2)
Tasse (3)
Kugel (4)
Giftflasche (5)
Männchen (6)
Flasche (7)
Diamant (8)

5. Möglichkeit:
Schuh (1)
Giftflasche (2)
Kelch (3)
Flasche (4)
Diamant (5)
Kugel (6)
Tasse (7)
Männchen (8)

6. Möglichkeit:
Männchen (1)
Kugel (2)
Diamant (3)
Schuh (4)
Flasche (5)
Kelch (6)
Giftflasche (7)
Tasse (8)

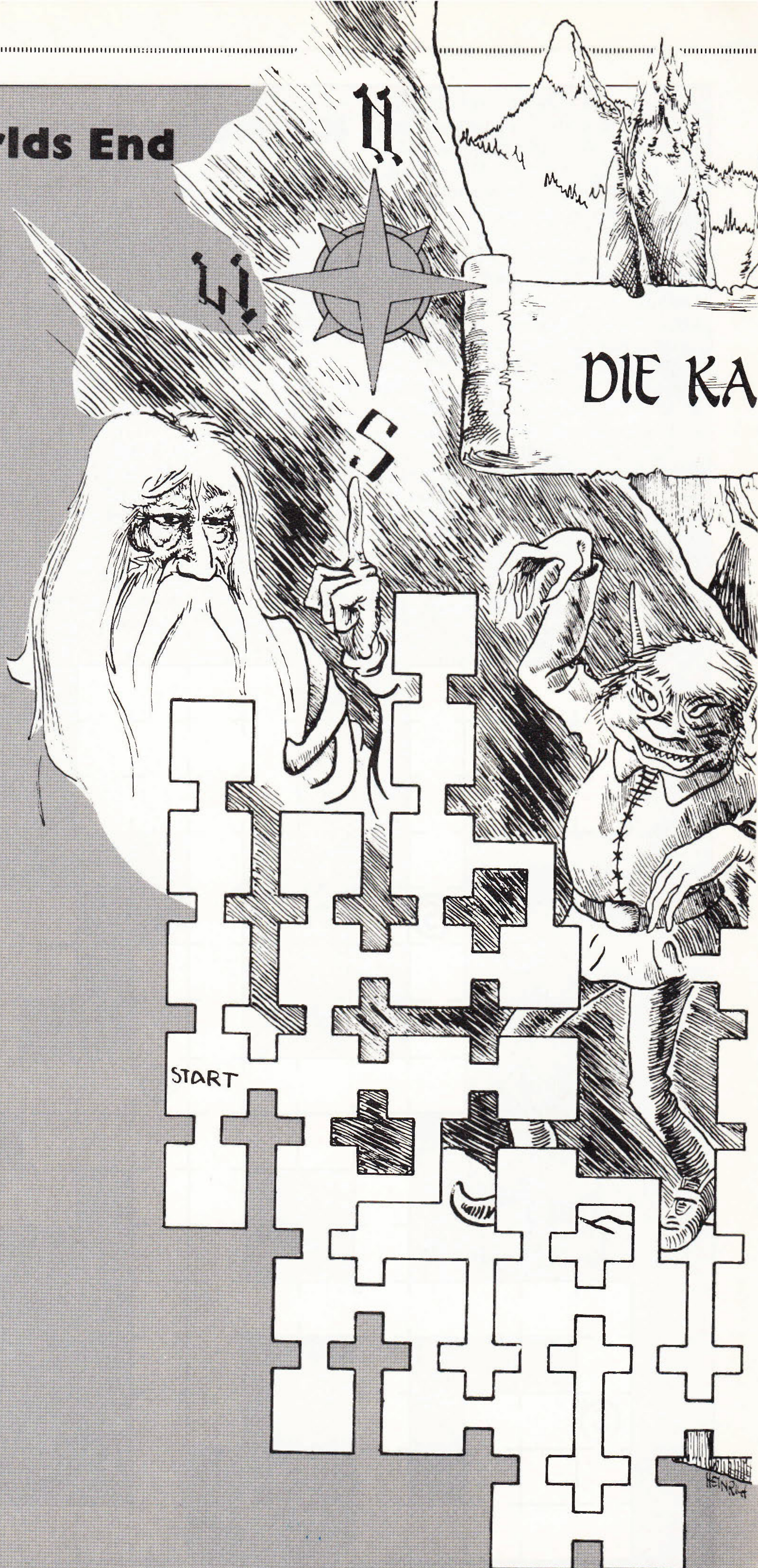
7. Möglichkeit:
Flasche (1)
Tasse (2)
Kugel (3)
Diamant (4)
Kelch (5)
Giftflasche (6)
Männchen (7)
Schuh (8)

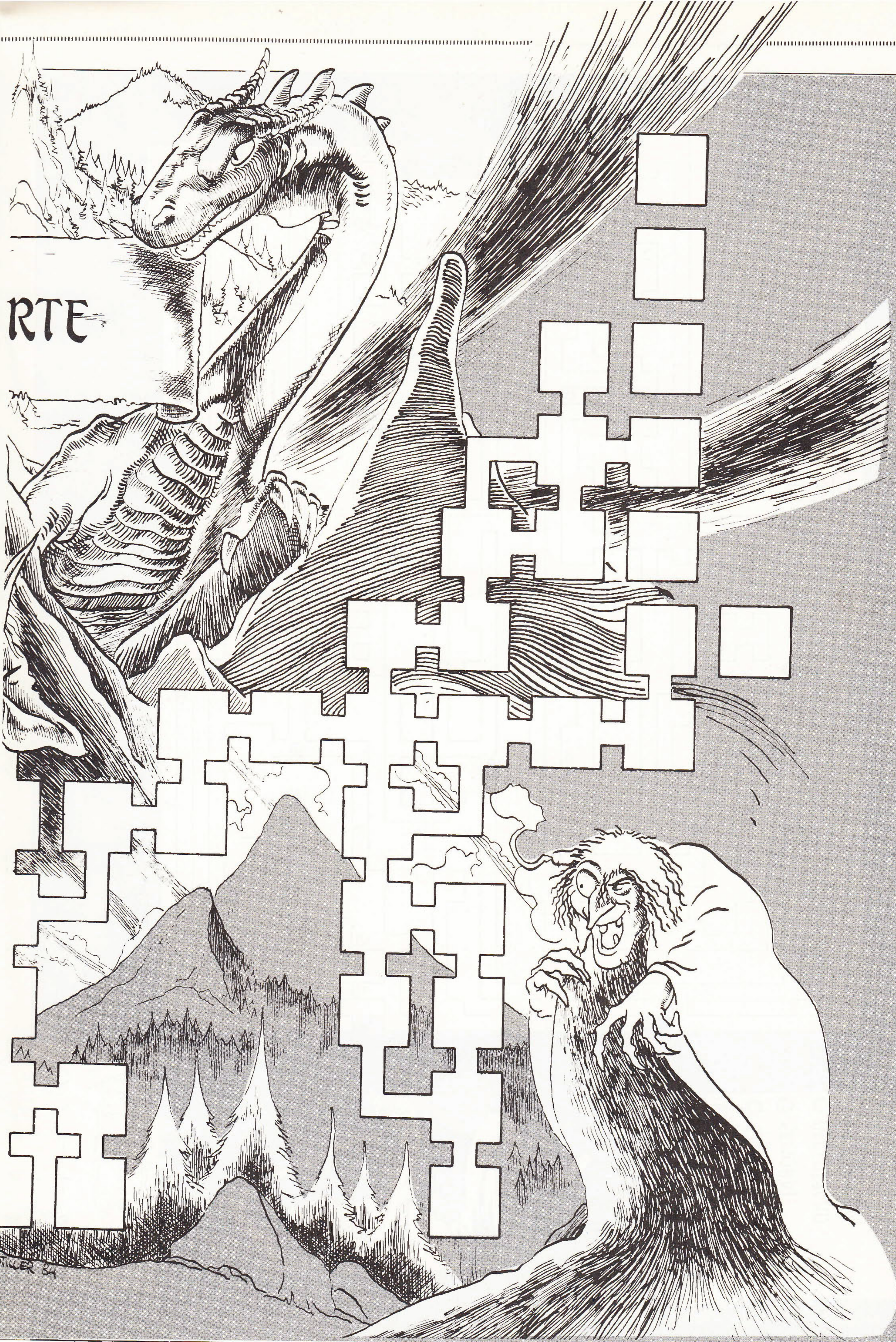
8. Möglichkeit:
Tasse (1)
Kelch (2)
Flasche (3)
Männchen (4)
Schuh (5)
Diamant (6)
Kugel (7)
Giftflasche (8)



Forest at Worlds End

Sowohl von der Story, wie auch von der Machart lehnt sich „Forest at Worlds End" ein wenig an den Hobbit an.
Der Spieler erhält die ehrenvolle Aufgabe, eine schöne Prinzessin aus den Klauen eines bösen Zauberers zu befreien. Doch bevor man dem Finsterling entgegentreten kann, müssen zuerst einmal einige andere Aufgaben gelöst werden.
Diese Karte erleichtert Ihnen, auch ohne weitere Angaben, das Spielen ungemein. Machen Sie sich eine Fotokopie davon und tragen Sie dann in diese Kopie alle Ihrer Meinung nach wichtigen Informationen ein. Sie werden sehen, vom guten Kartographieren hängt beim Abenteuern einiges ab.




RTE
STILLER 84

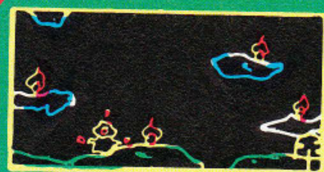
WILLOW PATTERN

KEY

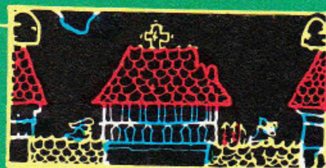


LEGENDE WILLOW PATTERN

- ✲ START
- ☠ SAMURAI
- ○ GEGENSTAND
- ♛ PRINZESSIN

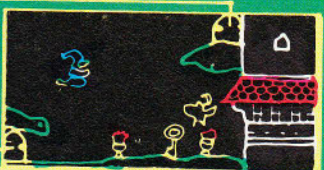

ABOVE THE WORLD


NEAR THE HIDEOUT


ABOVE THE HIDEOUT

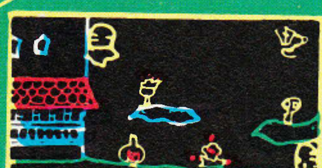

ABOVE THE HIDEOUT

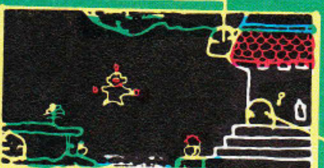

NEAR THE HIDEOUT


NEAR THE HIDEOUT


IN THE ARMORY


ABOVE THE MAIN HALL

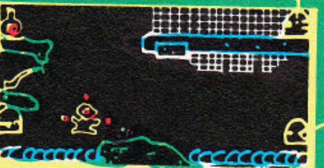

NEAR THE HIDEOUT

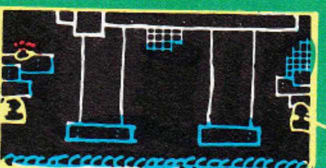

AT THE HIDEOUT ENTRANCE


NEAR THE NECROMANDER

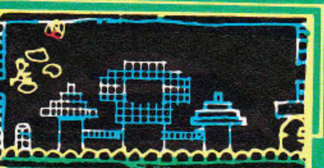

IN THE MAIN HALL (NECROMANDER)


NEAR THE HIDEOUT

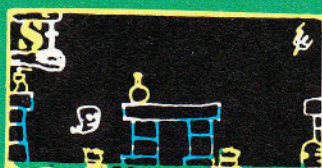

IN THE DESERTED CELLAR


UNDER THE MAIN HALL


UNDER THE MAN HALL


JUST IN TIME

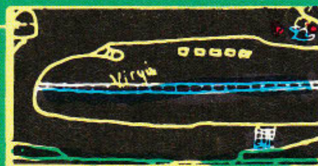
SORCERY+


AT STONEHENGE

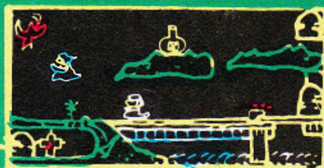

NEAR STONEHANGE

4.





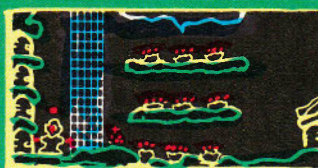
AT GATWICK ?



ON THE BRIDGE



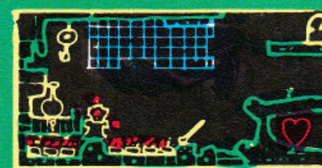
ON THE BRIDGE



AT THE MAIN GATE



IN THE LABYRINTH



UNDER THE GATEHOUSE



OUTSIDE THE CASTLE



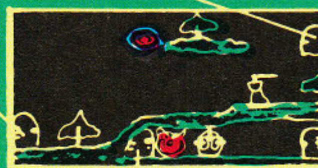
IN THE CASTLE



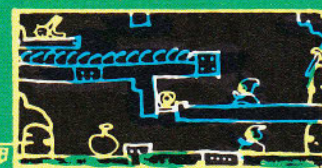
IN THE WASTELANDS



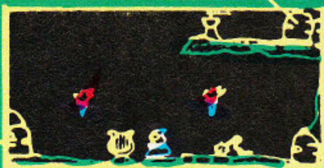
IN THE WASTELANDS



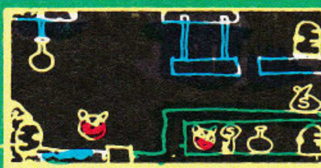
IN THE WASTELANDS



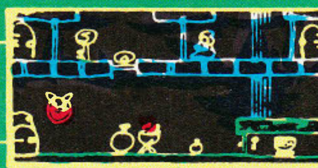
IN THE DUNGEONS



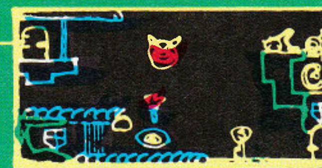
NEAR STONEHANGE



AT THE TUNNEL MOUTH



AT THE TUNNEL MOUSE



AT THE TUNNEL MOUTH

4.


IN THE BELFRY


UNDER THE BELFRY


NEAR THE GATEHOUSE


NEAR THE GATEHOUSE


NEAR THE PALACE


IN THE GATEHOUSE


AT SANCTUARY


NEAR THE PALACE


NEAR THE PALACE


UNDER SANJUARY


AT TH WATEFALL


ABOVE THE CHATEAU


IN THE CASTLE


OUTSIDE THE CASTLE


AT THE WATERFALL


NEAR THE CHATEAU


IN THE DUNGEONS

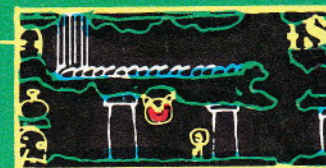

IN THE STRONGROOM


IN THE TUNNEL


IN THE TUNNEL

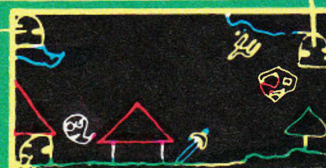

IN THE TUNNEL





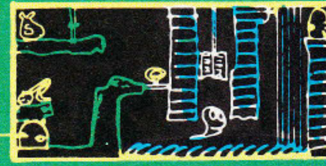


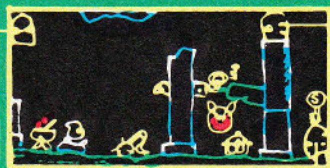


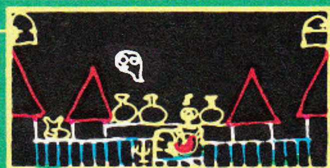








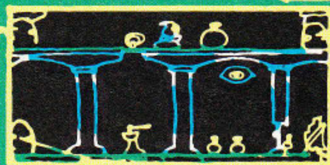




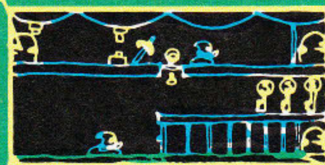


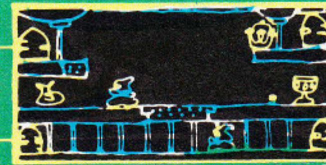












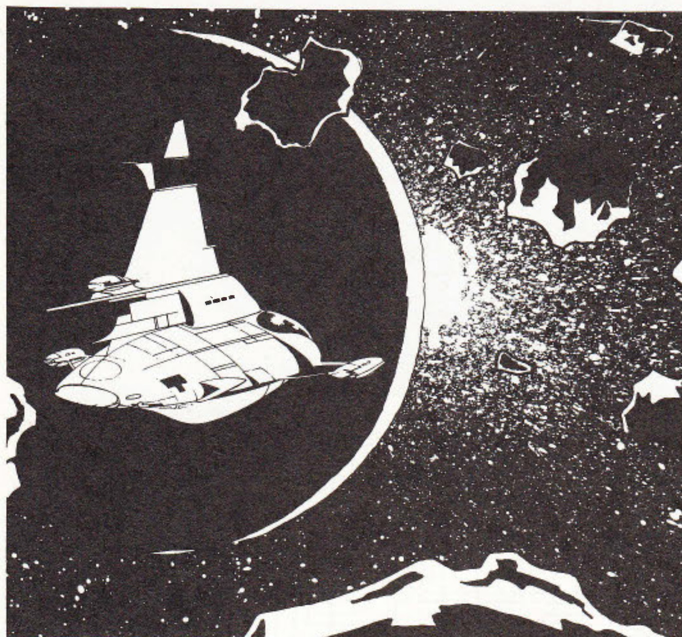




Diese detailreiche Sorcery-Karte stammt aus der Feder von Herrn v. Merten aus Düsseldorf.

SABRE WULF

Calactic Conflict



"Galactic Conflict" wird von einem Spieler gegen den Computer, im folgenden "Feind" genannt, gespielt.
Beide Parteien haben bis zu 20 Schiffe zur Verfügung, die sie aufeinanderhetzen sollen. Der Feind steuert seine Schiffe mit verschiedenen Strategien: einige verhalten sich aggressiv, andere abwartend oder Fallen stellend.
Der Spieler schickt nun seine Schiffe aus, um den Feind restlos auszurotten; dazu hat er zwei Waffen:

1. Die Laserkanone. Je nach abgeschossener Energieladung und Trefferdistanz verändert sich der erzeugte Schaden. Dieser reicht jedoch oft nicht aus, um mit einem einzigen Schiff ein einzelnes Feindschiff abzuballern. Daher gibt es:
2. die Raum-Minen. Abhängig von der Entfernung (möglicherweise flüchtet der Feind, nachdem man die Mine gelegt hat) entsteht mittelschwerer bis schwerer Schaden. Bei einem Volltreffer bleibt nichts vom getroffenen Objekt übrig.

Apropos Objekt: derer gibt es zwei:

1. Die Planeten. Dort kann der Spieler Energie nachtanken und, wenn er keine mehr hat, neue Minen besorgen. Der Feind kann lediglich Energie nachtanken. Planeten sind im allgemeinen nicht so einfach zu vernichten wie
2. die Raumbasen. Dort kann man zwar auch Energie abholen, aber diese Menge ist kleiner als bei einem Planeten. Dafür darf der Feind sie nicht benutzen. Dieser revanchiert sich seinerseits wiederum durch Attacken gegen Basen.

   Sind alle der etwas empfindlichen Basen abgeschossen, ist das Imperium, welches der Spieler verteidigen sollte, nicht mehr vorhanden.

Der Spieler kann Informationen über eigene Schiffe abrufen und mit dem Password "ENMSTR311671218" feindliche Informationen abzapfen.

Ein-/Ausgabeformate:
Der Ursprungspunkt (0;0) einer jeden Anzeige liegt in der linken oberen Ecke. Die auf dem oberen Teil des Bildschirms dargestellte Galaxis ist genau 20000 mal 10000 Einheiten groß; folglich hat jeder einzelne Sektor eine Größe von 1000 mal 1000 Einheiten. Eine Raumeinheit zu durchfliegen kostet exakt eine Energieeinheit.
Der in der Mitte unten dargestellte Sektor bezieht sich auf das jeweils mit einer Zahlentaste angewählte aktuelle Schiff. Dabei wird das Schiff in der Mitte gezeichnet; d.h., es wird nicht immer genau ein ganzer Sektor dargestellt, sondern es sind auch noch Teile benachbarter Sektoren aufgezeichnet. Zudem ist der Anzeigebereich etwas kleiner als ein voller Sektor.
Bei der Eingabe von Winkeln (beim Schießen oder Navigieren) ist zu beachten, daß rechts 0 Grad ist und gegen den Uhrzeigersinn gezählt wird.
Phasenaufteilung:
Da nicht sämtliche befohlenen Aktionen sofort ausgeführt werden, ist der folgende Phasenplan von Wichtigkeit:

1. Eingabe (!) von Navigation
   Legen von Minen
   Schießen
2. Bewegungen werden ausgeführt (Spieler und Feind)
3. Minen explodieren

Wenn der Benutzer eine Demonstration der verwendeten Shapes wünscht, wähle er bei der Eingabe des Schwierigkeitsgrades die Option CUSTOM (engl. "benutzerbestimmt") und gebe per Hand für jeden Typ die entsprechenden Anzahlen ein. Dabei erscheinen jeweils die Shapes.
Auf der Karte erscheinen alle Typen mit ihren Anfangsbuchstaben.

(K.J. Wolf)

```
10 ´(C)01/1986  written by K.J.Wolf          [2806]
20 ´REMs werden nicht angesprungen!          [3207]
30 CLEAR                                     [217]
40 TROFF                                     [431]
50 DEG                                       [133]
60 ON BREAK CONT                             [323]
70 ON ERROR GOTO 5480                        [1522]
80 ENV 1,15,-1,10                            [704]
90 ENV 2,4,-2,5,7,-1,50                      [1394]
100 ENV 3,1,0,0                              [477]
110 ENT 2,1,0,0                              [786]
120 ENT 3,70,-10,5                           [795]
130 DEFINT a-d,f-o,q-z                       [822]
140 emovmin=300                              [844]
150 rnchgstr!=0.02´probability of str chan [3437]
ge (0-1)
160 BORDER 1                                 [1011]
170 GOSUB 3750                               [1035]
180 MODE 2                                   [513]
190 INK 0,0                                  [53]
200 INK 1,23                                 [59]
210 CLS                                      [91]
220 DEF FNr(x)=INT(RND*x)                    [1736]
230 DEF FNz(n)=-(n<1)-n*(n>=1)               [758]
240 DEF FNq!(n)=n*n                          [364]
250 DEF FNm!(n!,a!)=n!-INT(n!/a!)*a!         [1482]
260 DIM px(4,20),py(4,20),n(4),lt(30),ln(3 [5893]
0),r$(19,9),e(4,20),tm(1),m(10),mvd(10),mv
v(10),str(20),n$(4),st$(2),ff(10)
270 FOR i=1 TO FNm!(TIME,3900)               [1612]
280 j=RND                                    [539]
290 NEXT                                     [350]
300 ´RANDOMIZE TIME does not function prop [3317]
erly
310 ORIGIN 316,135,160,472,32,239            [1286]
320 CLG 0                                    [222]
330 WINDOW #0,21,60,24,25                    [1303]
340 PAPER 0                                  [816]
350 PEN 1,0                                  [588]
360 CLS                                      [91]
370 WINDOW #1,1,80,1,10                      [983]
380 PAPER #1,0                               [1308]
390 PEN #1,1,0                               [606]
400 CLS #1                                   [373]
410 WINDOW #2,1,20,12,25                     [1534]
420 PAPER #2,0                               [1080]
430 PEN #2,1,0                               [748]
440 CLS #2                                   [372]
450 WINDOW #3,61,80,11,25                    [1491]
460 PAPER #3,0                               [1300]
```

```
470 PEN #3,1,0                              [700]
480 CLS #3                                  [371]
490 WINDOW #4,1,20,11,11                    [1282]
500 PAPER #4,0                              [1072]
510 PEN #4,1,0                              [590]
520 CLS #4                                  [386]
530 ´                                       [117]
540 PRINT #3,"M-MINE LEGEN"                 [1272]
550 PRINT #3,"0.9-SHIFF NR."                [1529]
560 PRINT #3,"D-DOCKEN"                     [765]
570 PRINT #3,"N-NAVIGIEREN"                 [974]
580 PRINT #3,"F-FEUERN"                     [401]
590 PRINT #3,"I-INFO"                       [685]
600 PRINT #3,"E-ENDE ZUG"                   [1045]
610 DATA "PLANET","BASIS","IHR SCHIFF","RA  [3390]
UMMINE","FEINDL.SCHIFF"
620 DATA "VERTEIDIGUNG","NOTFALL","NORMAL"  [2111]
630 RESTORE 610                             [745]
640 FOR i=0 TO 4                            [477]
650 READ n$(i)                              [487]
660 NEXT                                    [350]
670 FOR i=0 TO 2                            [471]
680 READ st$(i)                             [1023]
690 NEXT                                    [350]
700 PRINT "GALACTIC CONFLICT V1.1d"         [1307]
710 INPUT "UHRZEIT: Std,Min,Sek:",hours,mi  [2628]
nutes,seconds
720 IF hours<0 OR minutes<0 OR seconds<0 O  [5440]
R hours>23 OR minutes>59 OR seconds>59 THE
N 710
730 EVERY 50,0 GOSUB 760                    [1738]
740 GOTO 820                                [320]
750 ´                                       [117]
760 seconds=seconds+1                       [1378]
770 IF seconds>59 THEN seconds=0:minutes=m  [8509]
inutes+1:IF minutes>59 THEN minutes=0:hour
s=hours+1:SOUND 2,379,0,15,1:IF hours>23 T
HEN hours=0
780 LOCATE #4,1,1                           [364]
790 PRINT #4,USING "##:##:##";hours,minute  [2216]
s,seconds;
800 RETURN                                  [555]
810 ´                                       [117]
820 GOSUB 5530                              [873]
830 FOR i=0 TO 4                            [477]
840 IF i<>3 THEN FOR j=1 TO n(i):ON i+1 GO  [4415]
TO 850,870,890,920,920 ELSE 970
850 e(i,j)=50000                            [742]
860 GOTO 940                                [312]
870 e(i,j)=25000                            [906]
880 GOTO 940                                [312]
890 e(i,j)=6000+FNr(5500)                   [1240]
900 m(j)=4                                  [312]
910 GOTO 940                                [312]
920 e(i,j)=32500                            [562]
930 str(j)=FNr(5)                           [1542]
940 px(i,j)=FNr(20000)                      [596]
950 py(i,j)=FNr(10000)                      [1016]
960 NEXT j                                  [370]
970 NEXT i                                  [375]
980 tm(0)=20000                             [710]
990 tm(1)=10000                             [663]
1000 ´                                      [117]
1010 LOCATE #3,1,9                          [691]
1020 PRINT #3,"MENUE "                      [1008]
1030 GOSUB 4660                             [1061]
1040 GOSUB 3560                             [977]
1050 GOSUB 4460                             [829]
1060 GOSUB 1620                             [855]
1070 LOCATE #3,1,9                          [691]
1080 PRINT #3,"ZUG    "                     [823]
1090 GOSUB 2430                             [861]
1100 LOCATE #3,1,9                          [691]
1110 PRINT #3,"MINEN "                      [962]
1120 GOSUB 2940                             [895]
1130 GOSUB 3560                             [977]
1140 LOCATE #3,1,9                          [691]
1150 PRINT #3,"FEUERN"                      [981]
1160 GOSUB 3170                             [959]
1170 GOSUB 3560                             [977]
1180 GOTO 1010                              [345]
1190 ´*********************************** [2371]
***********************************
1200 ´**                              Anze [554]
ige                                 **
1210 ´*********************************** [2371]
***********************************
1220 IF refresh=0 THEN RETURN ELSE status= [3635]
0:IF refresh=99 THEN 1480
1230 WHILE e(2,ship)<=0                     [2088]
1240 ship=(ship+1)*-(ship<n(2))-(ship>=n(2 [1998]
))
1250 WEND                                   [390]
1260 FOR i=0 TO 19                          [532]
1270 FOR j=0 TO 9                           [723]
1280 r$(i,j)=""                             [173]
1290 NEXT j,i                               [396]
1300 n=0                                    [233]
1310 CLG 0                                  [222]
1320 xs=px(2,ship)\1000                     [1428]
1330 ys=py(2,ship)\1000                     [1412]
1340 FOR i=0 TO 4                           [477]
1350 IF n(i)=0 THEN 1400 ELSE FOR j=1 TO n [3106]
(i):IF e(i,j)<=0 THEN 1390
1360 IF refresh=88 THEN 1380 ELSE x0=px(i, [2831]
j)\1000:y0=py(i,j)\1000
1370 IF LEN(r$(x0,y0))<4 AND (i<3 OR r$(x0 [8475]
,y0)<>"") THEN IF i<>2 THEN r$(x0,y0)=r$(x
0,y0)+MID$("PBYMF",i+1,1) ELSE r$(x0,y0)=r
$(x0,y0)+RIGHT$(STR$(j-1),1)
1380 IF ABS(px(2,ship)-px(i,j))<=500 AND A [15013]
BS(py(2,ship)-py(i,j))<=500 THEN x=(px(i,j
)-px(2,ship))*0.16:y=-(py(i,j)-py(2,ship))
*0.104:status=status OR (i=4):ON i+1 GOSUB
 6970,7070,7210,7260,7340´range instead of
 sector!!
1390 NEXT j                                 [370]
1400 NEXT i                                 [375]
1410 IF refresh=88 THEN 1480                [1430]
1420 CLS #1                                 [373]
1430 FOR i=0 TO 19                          [532]
1440 FOR j=0 TO 9                           [723]
1450 LOCATE #1,i*4+1,j+1                    [648]
1460 IF r$(i,j)="" THEN PRINT #1,"."; ELSE [2454]
 PRINT #1,r$(i,j);
1470 NEXT j,i                               [396]
1480 CLS #2                                 [372]
1490 PRINT #2,"Shiff Nr.:";USING "#";ship- [3125]
1
1500 PRINT #2,"Energie :";USING "#####,";e [1993]
(2,ship)
1510 PRINT #2,"Position:";USING "##/#";px( [4141]
2,ship)\1000,py(2,ship)\1000
1520 PRINT #2,"Pos.Sek.: ";USING "#/#";(px [4035]
(2,ship) MOD 1000)\100,(py(2,ship) MOD 100
0)\100
1530 PRINT #2,"Minen    :";USING "#";m(ship [2187]
)
1540 IF status THEN status=0 ELSE IF e(2,s [5675]
hip)<4000 THEN status=1 ELSE status=2
1550 LOCATE #2,1,8                          [641]
1560 PRINT #2,st$(status)                   [1510]
1570 refresh=0                              [848]
1580 RETURN                                 [555]
1590 ´*********************************** [2371]
***********************************
1600 ´**                     Ziehen/Feuern [1964]
/Minen legen                        **
1610 ´*********************************** [2371]
***********************************
1620 GOSUB 1220                             [871]
1630 PRINT "Hauptmenue - waehlen Sie eine  [2762]
Funktion"
1640 CLEAR INPUT                            [184]
1650 ´ CLRINP kann weggelassen werden      [2379]
1660 a$=INKEY$                              [278]
1670 IF a$="" THEN 1660                     [715]
1680 a$=LOWER$(a$)                          [985]
1690 refresh=a$="i" OR a$="m" OR a$="f"     [2389]
1700 CLS                                    [91]
1710 IF (a$="0" OR VAL(a$)<>0) AND VAL(a$) [6546]
<n(2) THEN refresh=88:ship=VAL(a$)+1:GOTO
1620
1720 IF a$="i" THEN GOSUB 4920:GOTO 1620   [2303]
1730 IF a$="m" THEN GOSUB 1760 ELSE IF a$= [8827]
"d" THEN GOSUB 1850 ELSE IF a$="n" THEN GO
SUB 2050 ELSE IF a$="f" THEN GOSUB 2130 EL
SE IF a$="e" THEN CLS #1:CLG 0:CLS #2:RETU
RN
1740 GOTO 1620                              [343]
1750 ´min                                   [582]
1760 PRINT "MINE LEGEN - SIND SIE SICHER?? [2715]
"
1770 a$=INKEY$                              [278]
1780 IF a$="" THEN 1770                     [717]
1790 CLS                                    [91]
1800 a$=LOWER$(a$)                          [985]
1810 IF a$<>"j" THEN RETURN                 [1744]
```

```
1820 IF m(ship)>0 THEN m(ship)=m(ship)-1:n [9023]
(3)=n(3)+1:px(3,n(3))=px(2,ship):py(3,n(3)
)=py(2,ship):e(3,n(3))=10 ELSE PRINT "kein
e Minen":GOSUB 4810
1830 RETURN                                 [555]
1840 'dock                                  [743]
1850 PRINT "DOCKEN mit S)chiff / P)lanet / [4027]
 B)asis"
1860 a$=INKEY$                              [278]
1870 IF a$="" THEN 1860                     [754]
1880 CLS                                    [91]
1890 a$=LOWER$(a$)                          [985]
1900 i=-1                                   [453]
1910 IF a$="s" THEN i=2 ELSE IF a$="p" THE [4140]
N i=0 ELSE IF a$="b" THEN i=1
1920 IF i=-1 THEN PRINT "Fehler":GOSUB 481 [2959]
0:RETURN
1930 k=0                                    [189]
1940 FOR j=1 TO n(i)                        [1036]
1950 IF SQR(FNq!(px(2,ship)-px(i,j))+FNq!( [4820]
py(2,ship)-py(i,j)))<200 AND e(i,j)>0 THEN
 k=j
1960 NEXT                                   [350]
1970 IF k=0 THEN PRINT "Zu weit!":GOSUB 48 [2159]
10:RETURN
1980 refresh=99                             [989]
1990 GOSUB 4460                             [829]
2000 GOSUB 4460                             [829]
2010 IF i=2 THEN PRINT "DOCK OK. Uebertrag [12745]
ungsenergie?";:GOSUB 6160:e=INT(tx):CLS:IF
 e=0 THEN RETURN ELSE IF e<1 OR e>e(2,k)-2
50 OR e+e(2,ship)>tm(0) THEN 2010 ELSE e(2
,ship)=e(2,ship)+e:e(2,k)=e(2,k)-e-150:RET
URN
2020 IF i=0 AND m(ship)<=0 THEN m(ship)=1: [3685]
PRINT "1 Mine geladen":GOSUB 4810
2030 IF tm(i)<e(2,ship)+3500 THEN PRINT "P [7546]
otential zu hoch":GOSUB 4810:RETURN ELSE e
(2,ship)=tm(i):RETURN
2040 'nav                                   [138]
2050 PRINT "NAV - Richtung:Winkel?";        [3151]
2060 GOSUB 6160                             [907]
2070 mvv(ship)=tx                           [649]
2080 PRINT "NAV - Entfernung?";             [1775]
2090 GOSUB 6160                             [907]
2100 mvd(ship)=tx                           [789]
2110 IF mvv(ship)<0 OR mvv(ship)>360 OR mv [5197]
d(ship)<0 THEN 2050 ELSE CLS:RETURN
2120 'fire                                  [380]
2130 IF ff(ship) THEN PRINT "Sie haben ber [4943]
eits gefeuert":GOSUB 4810:RETURN ELSE ff(s
hip)=-1
2140 PRINT "FEUERN - Richtung:Winkel?";     [3472]
2150 GOSUB 6160                             [907]
2160 ia=tx                                  [206]
2170 PRINT "FEUERN - Energie?";             [2679]
2180 GOSUB 6160                             [907]
2190 ip=tx                                  [231]
2200 IF ip=0 THEN RETURN ELSE IF ia<0 OR i [5368]
a>360 OR ip<1 OR ip>e(2,ship) THEN 2130
2210 e(2,ship)=e(2,ship)-ip                 [848]
2220 d0=32000                               [730]
2230 k0=601                                 [362]
2240 FOR i=0 TO 4                           [477]
2250 IF n(i)<=0 OR i=3 THEN 2290 ELSE FOR  [5145]
j=1 TO n(i):IF e(i,j)<=0 OR (i=2 AND j=shi
p) THEN 2280
2260 k=SQR(FNq!(px(2,ship)-px(i,j))+FNq!(p [2540]
y(2,ship)-py(i,j)))
2270 IF k<>0 AND k<600 THEN x=px(2,ship)+k [9632]
*COS(ia):y=py(2,ship)-k*SIN(ia):d=SQR(FNq!
(x-px(i,j))+FNq!(y-py(i,j))):IF d<d0 THEN
k0=k:d0=d:i0=i:j0=j:x0=x:y0=y
2280 NEXT j                                 [370]
2290 NEXT i                                 [375]
2300 MOVE 0,0                               [483]
2310 IF d0<601 THEN DRAW (x0-px(2,ship))*0 [6301]
.16,-(y0-py(2,ship))*0.104,1,1 ELSE DRAWR
COS(ia)*200,SIN(ia)*200,1,1
2320 GOSUB 4500                             [837]
2330 IF d0<250 THEN PRINT "GETROFFEN (JUBE [8797]
L,JUBEL)!":e(i0,j0)=e(i0,j0)-110*ip/FNz(d0
):GOSUB 4540:IF e(i0,j0)<=0 THEN GOSUB 458
0:GOTO 2360 ELSE 2360
2340 PRINT "DANEBEN..."                     [2215]
2350 ' HEUTE SCHON DIE TAEGLICHE DOSIS FLA [3753]
SCHBIER GEHABT?
2360 IF d0<32000 THEN PRINT "Relative Entf [4685]
ernung:";CINT(d0)
2370 GOSUB 4810                             [960]
2380 GOSUB 4610                             [1063]
2390 RETURN                                 [555]
2400 '************************************ [2371]
***********************************
2410 '**                      Ziehen/feind [1651]
1. Manoever                            **
2420 '************************************ [2371]
***********************************
2430 FOR i=1 TO n(2)                        [426]
2440 IF mvd(i)=0 OR e(2,i)<=0 THEN 2530    [1993]
2450 mvd(i)=mvd(i)+FNr(mvd(i)\75)-FNr(mvd( [2578]
i)\75)
2460 IF mvd(i)>e(2,i) THEN PRINT "Shiff Nr [7200]
.";i-1;"ohne Energie":PRINT "Aktion abgebr
ochen";:GOSUB 4810:GOTO 2530
2470 e(2,i)=e(2,i)-mvd(i)                   [731]
2480 x=px(2,i)+mvd(i)*COS(mvv(i))           [2946]
2490 y=py(2,i)-mvd(i)*SIN(mvv(i))           [2146]
2500 IF x<=0 OR y<=0 OR x>=20000 OR y>=100 [14297]
00 THEN PRINT "Fehler: Shiff Nr.";i-1;"ver
laesst Galaxis":PRINT "alle Maschinen voll
e Kraft f.Raumsprung";:e(2,i)=INT(e(2,i)/1
0):x=FNr(20000):y=FNr(10000):GOSUB 4810
2510 px(2,i)=x                              [411]
2520 py(2,i)=y                              [142]
2530 NEXT i                                 [375]
2540 '                                      [117]
2550 FOR i=1 TO n(4)                        [438]
2560 IF RND<rnchgstr! THEN str(i)=FNr(5)    [1380]
2570 IF e(4,i)<=0 OR RND>rnemov! THEN 2790 [4466]
 ELSE ON str(i)+1 GOTO 2580,2620,2660,2700
,2740,2750
2580 j=-1                                   [461]
2590 k=2                                    [187]
2600 GOSUB 2810                             [849]
2610 GOTO 2790                              [317]
2620 j=1                                    [411]
2630 k=1                                    [192]
2640 GOSUB 2810                             [849]
2650 GOTO 2790                              [317]
2660 j=1                                    [411]
2670 k=0                                    [189]
2680 GOSUB 2810                             [849]
2690 GOTO 2790                              [317]
2700 j=1                                    [411]
2710 k=2                                    [187]
2720 GOSUB 2810                             [849]
2730 GOTO 2790                              [317]
2740 GOTO 2790                              [317]
2750 j=1                                    [411]
2760 k=0                                    [189]
2770 GOSUB 2810                             [849]
2780 str(i)=FNr(5)                          [660]
2790 NEXT                                   [350]
2800 RETURN                                 [555]
2810 d=32000                                [754]
2820 FOR i0=1 TO n(k)                       [774]
2830 d0=SQR(FNq!(px(k,i0)-px(4,i))+FNq!(py [3568]
(k,i0)-py(4,i)))
2840 IF e(k,i0)>0 AND d0<d AND d0>emovmin* [1641]
j THEN d=d0:i1=i0
2850 NEXT                                   [350]
2860 IF d=32000 THEN str(i)=FNr(5):RETURN  [7124]
ELSE IF e(4,i)<1500 THEN str(i)=5:k=0:j=1:
GOTO 2810
2870 IF d>e(4,i) THEN IF str(i)=5 THEN str [4698]
(i)=FNr(5):RETURN ELSE str(i)=5:RETURN
2880 IF j=-1 THEN d0=FNr(500)+500:x0=-SGN( [11939]
px(k,i1)-px(4,i))*d0:y0=-SGN(py(k,i1)-py(4
,i))*d0:d=SQR(FNq!(x0)+FNq!(y0)):x0=x0+px(
4,i):y0=y0+py(4,i) ELSE x0=px(k,i1)+FNr(80
0)-400:y0=py(k,i1)+FNr(800)-400
2890 IF x0>0 AND y0>0 AND x0<20000 AND y0< [7204]
10000 THEN e(4,i)=e(4,i)-d:px(4,i)=x0:py(4
,i)=y0:IF k=0 AND e(4,i)<tm(0) THEN e(4,i)
=tm(0)
2900 RETURN                                 [555]
2910 '************************************ [2371]
***********************************
2920 '**                        Minen exp [1308]
lodieren                          **
2930 '************************************ [2371]
***********************************
2940 IF n(3)<=0 THEN RETURN ELSE FOR i=1 T [2915]
O n(3):IF e(3,i)<=0 THEN 3070
2950 xs=px(3,i)                             [1264]
2960 ys=py(3,i)                             [1162]
2970 PRINT "MINE EXPLODIERT IN QUADRANT "; [4323]
```

```
USING "##/#";xs\1000,ys\1000
2980 GOSUB 6090                               [833]
2990 GOSUB 4540                               [829]
3000 FOR j=0 TO 4                             [692]
3010 IF j=3 THEN 3050 ELSE FOR k=1 TO n(j)    [805]
3020 i0=SQR(FNq!(px(3,i)-px(j,k))+FNq!(py(   [2899]
3,i)-py(j,k)))
3030 IF i0<=350 AND e(j,k)>0 THEN e(j,k)=e   [5758]
(j,k)-2000000/FNz(i0):PRINT "Treffer! ";:I
F e(j,k)<=0 THEN GOSUB 4580
3040 NEXT k                                   [373]
3050 NEXT j                                   [370]
3060 PRINT                                    [361]
3070 GOSUB 4810                               [960]
3080 GOSUB 4610                              [1063]
3090 e(3,i)=0                                 [411]
3100 NEXT i                                   [375]
3110 n(3)=0                                   [420]
3120 CLG 0                                    [222]
3130 RETURN                                   [555]
3140 '*********************************** [2371]
**********************************
3150 '**                          feindl.   [1381]
 Feuer                          **
3160 '*********************************** [2371]
**********************************
3170 FOR k=1 TO n(4)                         [1522]
3180 IF e(4,k)<4000 THEN str(k)=5             [873]
3190 enm=0                                    [109]
3200 fp=0                                     [426]
3210 FOR i=0 TO 2                             [471]
3220 FOR j=1 TO n(i)                         [1036]
3230 IF e(i,j)>0 THEN d=SQR(FNq!(px(4,k)-p   [5693]
x(i,j))+FNq!(py(4,k)-py(i,j))):IF d<600 TH
EN ON i+1 GOSUB 3280,3300,3320:IF f THEN 3
260
3240 NEXT j,i                                 [396]
3250 IF enm>1 THEN str(k)=0                  [1229]
3260 NEXT k                                   [373]
3270 RETURN                                   [555]
3280 fp=-1                                    [489]
3290 RETURN                                   [555]
3300 IF str(k)=1 OR str(k)=4 THEN 3350       [1465]
3310 RETURN                                   [555]
3320 enm=enm+1                                [456]
3330 IF (str(k)=2 AND fp) OR str(k)=3 OR s   [3627]
tr(k)=4 THEN 3350
3340 RETURN                                   [555]
3350 IF RND>rnefir! THEN f=0:RETURN ELSE f   [1598]
=-1
3360 xs=px(i,j)                              [1384]
3370 ys=py(i,j)                              [1452]
3380 PRINT "feindl. Beschuss in ";USING "#   [3967]
#/#";xs\1000,ys\1000
3390 GOSUB 6090                               [833]
3400 MOVE 0,0                                 [483]
3410 DRAW (px(4,k)-xs)*0.16,-(py(4,k)-ys)*   [2946]
0.104,1,1
3420 GOSUB 4500                               [837]
3430 GOSUB 4540                               [829]
3440 ip=CINT(e(4,k)/10)                      [1277]
3450 e(4,k)=e(4,k)-ip                         [949]
3460 d=d/5-FNr(d/10)                         [1090]
3470 e(i,j)=e(i,j)-100*ip/FNz(d)             [1447]
3480 IF e(i,j)<=0 THEN GOSUB 4580            [1667]
3490 GOSUB 4810                               [960]
3500 GOSUB 4610                              [1063]
3510 CLG 0                                    [222]
3520 RETURN                                   [555]
3530 '*********************************** [2371]
**********************************
3540 '**                       Test: wer u   [1360]
eberlebt?                       ***
3550 '*********************************** [2371]
**********************************
3560 f=0                                      [392]
3570 FOR i=1 TO n(2)                          [426]
3580 f=f OR e(2,i)>0                         [1063]
3590 NEXT                                     [350]
3600 IF f=0 THEN n=REMAIN(0):MODE 0:LOCATE  [11136]
 1,12:SPEED INK 15,15:INK 2,3,12:PAPER 2:P
RINT "SIE VERLIEREN!!":PRINT "(alle Ihre S
hiffe":PRINT " vernichtet)":PAPER 0:PRINT:
PRINT:END
3610 f=0                                      [392]
3620 FOR i=1 TO n(4)                          [438]
3630 f=f OR e(4,i)>0                         [1328]
3640 NEXT                                     [350]
3650 IF f=0 THEN n=REMAIN(0):MODE 0:LOCATE   [9690]
 1,12:SPEED INK 15,15:INK 2,3,12:PAPER 2:P
RINT "SIE GEWINNEN!!!!":PRINT "(alle Feind
e":PRINT " zerstoert)":PAPER 0:PRINT:PRINT
:END
3660 f=0                                      [392]
3670 FOR i=1 TO n(1)                          [448]
3680 f=f OR e(1,i)>0                         [1055]
3690 NEXT                                     [350]
3700 IF f=0 THEN n=REMAIN(0):MODE 0:LOCATE  [10040]
 1,12:SPEED INK 15,15:INK 2,3,12:PAPER 2:P
RINT "SIE VERLIEREN!!":PRINT "(alle Basen"
:PRINT " verloren)":PAPER 0:PRINT:PRINT:EN
D
3710 RETURN                                   [555]
3720 '*********************************** [2371]
**********************************
3730 '**                             Hall   [509]
o!                              ***
3740 '*********************************** [2371]
**********************************
3750 SYMBOL AFTER 248                        [1416]
3760 SYMBOL 248,0,0,198,198,198,252,192,19   [2025]
2
3770 SYMBOL 249,8,16,124,198,198,124         [2170]
3780 SYMBOL 250,0,0,124,198,198,124          [2330]
3790 SYMBOL 251,0,0,198,102,108,240          [1943]
3800 SYMBOL 252,0,0,48,48,48,24              [1629]
3810 SYMBOL 253,0,0,230,110,126,103          [1636]
3820 SYMBOL 254,0,0,124,192,192,124,6,12     [2214]
3830 DATA 248,249,251,250,252,253,250,254    [1534]
3840 MODE 0                                   [507]
3850 INK 0,0                                   [53]
3860 INK 1,23                                  [59]
3870 PAPER 0                                  [816]
3880 PEN 1,0                                  [588]
3890 LOCATE 1,7                               [629]
3900 PRINT "  G A L A C T I C"               [1354]
3910 PRINT "  C O N F L I C T"               [1447]
3920 PRINT                                    [361]
3930 PRINT "    Version 1.1d"                [1699]
3940 PRINT "         (C)"                     [929]
3950 PRINT "       Jan  86"                   [587]
3960 PRINT "         b y"                     [763]
3970 PRINT "          ";                      [750]
3980 RESTORE 3830                             [777]
3990 FOR i=1 TO 8                             [462]
4000 READ j                                   [455]
4010 PRINT CHR$(j);                          [1164]
4020 NEXT                                     [350]
4030 LOCATE 1,24                              [728]
4040 PRINT "(A)nleitung oder"                [2226]
4050 PRINT"           (S)piel"                [769]
4060 FOR i=2 TO 15                            [610]
4070 INK i,0                                  [337]
4080 NEXT                                     [350]
4090 j=2                                      [414]
4100 FOR i=306 TO 400 STEP 10                 [992]
4110 MOVE 303,306                             [317]
4120 DRAW 0,i,j,0                             [434]
4130 j=j+1                                    [544]
4140 IF j>15 THEN j=2                         [839]
4150 NEXT                                     [350]
4160 FOR i=0 TO 640 STEP 20                   [555]
4170 MOVE 303,306                             [317]
4180 DRAW i,400,j                             [367]
4190 j=j+1                                    [544]
4200 IF j>15 THEN j=2                         [839]
4210 NEXT                                     [350]
4220 FOR i=400 TO 306 STEP -10               [1060]
4230 MOVE 303,306                             [317]
4240 DRAW 640,i,j                             [191]
4250 j=j+1                                    [544]
4260 IF j>15 THEN j=2                         [839]
4270 NEXT                                     [350]
4280 DIM a(15)                                [104]
4290 FOR i=2 TO 15                            [610]
4300 a(i)=(i-2)*2                             [536]
4310 NEXT                                     [350]
4320 a$=""                                    [388]
4330 WHILE a$=""                              [933]
4340 a$=INKEY$                                [278]
4350 a(1)=a(2)                                [551]
4360 FOR i=2 TO 15                            [610]
4370 INK i,a(i)                               [788]
4380 IF i<15 THEN a(i)=a(i+1)                [1383]
4390 NEXT                                     [350]
4400 a(15)=a(1)                               [619]
4410 WEND                                     [390]
4420 IF LOWER$(a$)="a" THEN 6260 ELSE RETU   [2627]
```

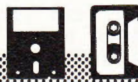
```
RN
4430 '************************************ [2371]
************************************
4440 '**                        Gong,Zap,Ba [1986]
ff,Flimmer                              **
4450 '************************************ [2371]
************************************
4460 SOUND 1,95,0,15,1                    [1547]
4470 WHILE SQ(1) AND &80                  [1109]
4480 WEND                                 [390]
4490 RETURN                               [555]
4500 SOUND 1,400,500,15,2,2,15            [1757]
4510 WHILE SQ(1) AND &80                  [1109]
4520 WEND                                 [390]
4530 RETURN                               [555]
4540 SOUND 1,700,350,15,3,3,15            [1559]
4550 WHILE SQ(1) AND &80                  [1109]
4560 WEND                                 [390]
4570 RETURN                               [555]
4580 BORDER 12,15                         [1189]
4590 SPEED INK 1,1                        [1280]
4600 RETURN                               [555]
4610 BORDER 1                             [1011]
4620 RETURN                               [555]
4630 '************************************ [2371]
************************************
4640 '**                  alles klaeren; E [2066]
nergieverbrauch                         **
4650 '************************************ [2371]
************************************
4660 refresh=-1                           [1053]
4670 FOR i=1 TO n(2)                      [426]
4680 mvd(i)=0                             [313]
4690 mvv(i)=0                             [377]
4700 ff(i)=0                              [485]
4710 e(2,i)=e(2,i)-100                    [960]
4720 NEXT                                 [350]
4730 FOR i=1 TO n(4)                      [438]
4740 e(4,i)=e(4,i)-100                    [592]
4750 NEXT                                 [350]
4760 ship=0                               [655]
4770 RETURN                               [555]
4780 '************************************ [2371]
************************************
4790 '**                            Tasten [1040]
druck                                   **
4800 '************************************ [2371]
************************************
4810 LOCATE #3,1,11                       [646]
4820 PRINT #3,"TASTENDRUCK!"              [1536]
4830 WHILE INKEY$=""                      [1095]
4840 WEND                                 [390]
4850 LOCATE #3,1,11                       [646]
4860 PRINT #3,"             "             [1263]
4870 CLS                                  [91]
4880 RETURN                               [555]
4890 '************************************ [2371]
************************************
4900 '**                            Info-S [1514]
eiten                                   **
4910 '************************************ [2371]
************************************
4920 CLG 0                                [222]
4930 CLS #1                               [373]
4940 CLS #2                               [372]
4950 CLS                                  [91]
4960 PRINT "Daten zum Drucker? "          [3205]
4970 a$=""                                [388]
4980 WHILE a$=""                          [933]
4990 a$=INKEY$                            [278]
5000 WEND                                 [390]
5010 CLS                                  [91]
5020 IF LOWER$(a$)="j" THEN ch=8 ELSE ch=1 [1184]
5030 IF (INP(&F500) AND 64)<>0 AND ch=8 TH [5582]
EN PRINT "Drucker nicht bereit!":GOTO 4960
5040 FOR j=1 TO n(2) STEP 2               [1193]
5050 FOR i=j TO j+1                       [1313]
5060 PRINT #ch,TAB((i-j)*40);USING "# ";i- [3381]
1;
5070 IF e(2,i)>0 THEN PRINT #ch,USING "### [9211]
##/#### #####, # ### #####,";px(2,i),py(2,
i),e(2,i),m(i),mvv(i),mvd(i); ELSE IF i<=n
(2) THEN PRINT #ch,"ZERSTOERT";
5080 NEXT                                 [350]
5090 NEXT                                 [350]
5100 PRINT #2,n$(2)                       [553]
5110 PRINT #2,"Daten-Liste zeigt:"        [1335]
5120 PRINT #2,"Position X/Y"              [1974]
5130 PRINT #2,"Energie Einheiten"         [1714]
5140 PRINT #2,"Anzahl Minen"              [1840]
5150 PRINT #2," NAV-Winkel &"             [575]
5160 PRINT #2," Entfernung"               [1170]
5170 PRINT "naechste Datenseite benoetigt [3461]
ein"
5180 PRINT "SysOp Password Stufe 1"       [2759]
5190 FOR i=0 TO 5000                      [439]
5200 NEXT                                 [350]
5210 CLS                                  [91]
5220 PRINT "Pass:";                       [1166]
5230 GOSUB 6160                           [907]
5240 pw$=tx$                              [326]
5250 CLS                                  [91]
5260 IF LOWER$(pw$)<>"enmstr311671218" THE [5912]
N PRINT "PASSWORD ERROR":GOSUB 4810:CLS #1
:CLS #2:RETURN
5270 CLS #1                               [373]
5280 CLS #2                               [372]
5290 FOR j=1 TO n(4) STEP 2               [1564]
5300 FOR i=j TO j+1                       [1313]
5310 PRINT #ch,TAB((i-j)*40);USING "## ";i [3142]
;
5320 IF e(4,i)>0 THEN PRINT #ch,USING "### [8001]
##/#### #####, #";px(4,i),py(4,i),e(4,i),s
tr(i); ELSE IF i<=n(4) THEN PRINT #ch,"ZER
STOERT";
5330 NEXT                                 [350]
5340 NEXT                                 [350]
5350 PRINT #2,n$(4)                       [565]
5360 PRINT #2,"Daten-Liste zeigt:"        [1335]
5370 PRINT #2,"Position X/Y"              [1974]
5380 PRINT #2,"Energie Einheiten"         [1714]
5390 PRINT #2,"STRATEGIE-CODE"            [1437]
5400 PRINT "Vertraulich! - bitte pruefen S [3222]
ie"
5410 GOSUB 4810                           [960]
5420 CLS #1                               [373]
5430 CLS #2                               [372]
5440 RETURN                               [555]
5450 '************************************ [2371]
************************************
5460 '**               Fehler-Falle   !Auf [1499]
passen mit RENUM!                       **
5470 '************************************ [2371]
************************************
5480 DI                                   [84]
5490 IF ERL=6230 THEN EI:RESUME 6160 ELSE [4740]
n=REMAIN(0):MODE 2:ON ERROR GOTO 0
5500 '************************************ [2371]
************************************
5510 '**                      User: Szenari [2822]
o festlegen                            ***
5520 '************************************ [2371]
************************************
5530 PRINT "Waehlen Sie den Schwierigkeits [5111]
grad:"
5540 PRINT "1(leicht)-4(unmoeglich)/5:eige [4217]
ne Werte"
5550 rnefir!=0.4                          [1302]
5560 a$=""                                [388]
5570 WHILE a$=""                          [933]
5580 a$=INKEY$                            [278]
5590 WEND                                 [390]
5600 IF a$<"1" OR a$>"5" THEN 5560        [2344]
5610 CLS                                  [91]
5620 ON VAL(a$) GOTO 5630,5690,5750,5810,5 [1776]
870
5630 rnemov!=0.3                          [1184]
5640 n(0)=20                              [420]
5650 n(1)=20                              [245]
5660 n(2)=5                               [411]
5670 n(4)=5                               [363]
5680 RETURN                               [555]
5690 rnemov!=0.4                          [1175]
5700 n(0)=15                              [425]
5710 n(1)=10                              [255]
5720 n(2)=10                              [399]
5730 n(4)=15                              [489]
5740 RETURN                               [555]
5750 rnemov!=0.65                         [1158]
5760 n(0)=7                               [265]
5770 n(1)=10                              [255]
5780 n(2)=10                              [399]
5790 n(4)=20                              [484]
5800 RETURN                               [555]
5810 rnemov!=0.775                        [1196]
5820 n(0)=3                               [269]
5830 n(1)=7                               [225]
5840 n(2)=10                              [399]
```

```
5850 n(4)=20                                    [484]
5860 RETURN                                     [555]
5870 FOR i=0 TO 4                               [477]
5880 IF i<>3 THEN x=0:y=0:ON i+1 GOSUB 697 [9523]
0,7070,7210,7260,7340:WHILE n(i)<1 OR n(i)
>20+10*(i=2):PRINT "(";n$(i);") Wie viele?
";:GOSUB 6160:n(i)=tx:WEND:CLG 0
5890 NEXT                                       [350]
5900 CLS                                        [91]
5910 PRINT "feindl.Aktivitaet (0=null-1=su [3506]
perhoch)"
5920 LOCATE 1,2                                 [614]
5930 PRINT SPACE$(19);                          [1175]
5940 LOCATE 1,2                                 [614]
5950 PRINT "Bewegung:";                         [1967]
5960 GOSUB 6160                                 [907]
5970 rnemov!=VAL(tx$)                           [1097]
5980 LOCATE 1,2                                 [614]
5990 PRINT SPACE$(19);                          [1175]
6000 LOCATE 1,2                                 [614]
6010 PRINT "Laserbenutzung:";                   [2179]
6020 GOSUB 6160                                 [907]
6030 rnefir!=VAL(tx$)                           [1849]
6040 IF rnemov!<0 OR rnefir!<0 OR rnemov!> [5165]
1 OR rnefir!>1 THEN 5920 ELSE CLS
6050 RETURN                                     [555]
6060 '************************************ [2371]
***********************************
6070 '**                         Anleitung (v [868]
on Hallo!)                        ***
6080 '************************************ [2371]
***********************************
6090 CLG 0                                      [222]
6100 FOR zi=0 TO 4                              [1045]
6110 IF n(zi)=0 THEN 6140 ELSE FOR zj=1 TO [4360]
 n(zi):IF e(zi,zj)<=0 THEN 6130
6120 IF ABS(xs-px(zi,zj))<=500 AND ABS(ys- [7460]
py(zi,zj))<=500 THEN x=(px(zi,zj)-xs)*0.16
:y=-(py(zi,zj)-ys)*0.104:ON zi+1 GOSUB 697
0,7070,7210,7260,7340
6130 NEXT                                       [350]
6140 NEXT zi                                    [415]
6150 RETURN                                     [555]
6160 tx$=""                                     [55]
6170 PRINT CHR$(143);                           [1025]
6180 x$=INKEY$                                  [1806]
6190 IF x$="" THEN 6180                         [1428]
6200 x$=UPPER$(x$)                              [1393]
6210 IF x$<="Z" AND x$>=" " AND LEN(tx$)<1 [4936]
6 THEN PRINT CHR$(8);x$;CHR$(143);:tx$=tx$
+x$:GOTO 6180
6220 IF x$=CHR$(127) AND tx$<>"" THEN PRIN [6605]
T CHR$(8);" ";STRING$(2,8);CHR$(143);:tx$=
LEFT$(tx$,LEN(tx$)-1):GOTO 6180
6230 IF x$=CHR$(13) THEN PRINT CHR$(8);" " [5071]
:tx=VAL(tx$):RETURN
6240 PRINT CHR$(7);                             [1175]
6250 GOTO 6180                                  [311]
6260 MODE 2                                     [513]
6270 WINDOW #1,1,80,1,1                         [995]
6280 WINDOW #0,1,80,2,25                        [1196]
6290 CLS                                        [91]
6300 CLS #1                                     [373]
6310 PRINT #1,"GALACTIC CONFLICT -Anleitun [2979]
g-"
6320 PRINT "ANLEITUNG/BESCHREIBUNG DER SPI [3046]
ELVORGAENGE:"
6330 PRINT "Das Spiel GALACTIC CONFLICT wi [5573]
rd von einem Spieler gegen den Computer,"
6340 PRINT "im folgenden 'Feind' genannt.  [3590]
"
6350 PRINT "Beide Parteien haben bis zu 20 [5379]
 Schiffe zur Verfuegung,"
6360 PRINT "die sie aufeinanderhetzen soll [8896]
en. Der Feind steuert seine Schiffe mit"
6370 PRINT "verschiedenen Strategien: eini [7279]
ge verhalten sich aggressiv, andere abwart
end"
6380 PRINT "oder Fallen stellend. Der Spie [7668]
ler schickt nun seine Schiffe aus, um den
Feind"
6390 PRINT "restlos auszurotten; dazu hat  [6357]
er zwei Waffen:"
6400 PRINT "1. Die Laserkanone. Je nach ab [8567]
geschossener Energieladung und Trefferdist
anz"
6410 PRINT "   veraendert sich der erzeugt [7437]
e Schaden. Dieser ist jedoch oft nicht aus
-"
6420 PRINT "   reichend um mit einem einzi [6366]
gen Schiff ein einzelnes Feindschiff abzu-
"
6430 PRINT "   ballern. Daher gibt es"      [2573]
6440 PRINT "2. Die Raumminen. Abhaengig vo [8166]
n der Entfernung (moeglicherweise fluechte
t der"
6450 PRINT "   Feind, nachdem man die Mine [8276]
 gelegt hat) entsteht mittelschwerer bis"
6460 PRINT "   schwerer Schaden. Bei einem [9618]
 Volltreffer bleibt nichts vom getroffenen
"
6470 PRINT "   Objekt uebrig."              [1332]
6480 PRINT "Apropos Objekt: Derer gibt es  [2955]
zwei:"
6490 PRINT "1. Die Planeten. Dort kann der [6623]
 Spieler Energie nachtanken und, wenn"
6500 PRINT "   er keine mehr hat, neue Min [8057]
en besorgen. Der Feind kann lediglich Ener
gie"
6510 PRINT "   nachtanken. Planeten sind i [5302]
m allgemeinen nicht so einfach zu vernicht
en wie"
6520 PRINT "2. Die Raumbasen. Dort kann ma [8449]
n zwar auch Energie abholen, aber diese Me
nge"
6530 PRINT "   ist kleiner als bei einem P [6866]
laneten. Dafuer darf der Feind sie nicht b
e-"
6540 PRINT "   nutzen. Dieser revanchiert  [7268]
seinerseits wiederum durch Attacken gegen
Basen."
6550 WHILE INKEY$=""                            [1095]
6560 WEND                                       [390]
6570 CLS                                        [91]
6580 PRINT "   Sind alle der etwas empfind [5140]
lichen Basen abgeschossen, ist das Imperiu
m,"
6590 PRINT "   welches der Spieler verteid [5168]
igen sollte, nicht mehr vorhanden."
6600 PRINT "Der Spieler kann Informationen [6694]
 ueber eigene Schiffe abrufen und mit dem"
6610 PRINT "Password 'ENMSTR311671218' fei [5540]
ndliche Informationen abzapfen."
6620 PRINT "Ein-/Ausgabeformate:"           [2314]
6630 PRINT "Der Ursprungspunkt (0;0) einer [7751]
 jeden Anzeige liegt in der linken oberen"
6640 PRINT "Ecke. Die auf dem oberen Teil  [6502]
des Bildschirms dargestellte Galaxis ist g
enau"
6650 PRINT "20000 mal 10000 Einheiten gros [6748]
s; folglich hat jeder einzelne Sektor eine
 "
6660 PRINT "Groesse von 1000 mal 1000 Einh [7755]
eiten. Eine Raumeinheit zu durchfliegen ko
stet"
6670 PRINT "exakt eine Energieeinheit. Der [7368]
 in der Mitte unten dargestellte Sektor"
6680 PRINT "bezieht sich auf das jeweils m [8242]
it einer Zahlentaste angewaehlte aktuelle
Schiff."
6690 PRINT "Dabei wird das Schiff in der M [9092]
itte gezeichnet; d.h. es wird nicht immer
"
6700 PRINT "genau ein ganzer Sektor darges [4841]
tellt, sondern es sind auch noch Teile"
6710 PRINT "benachbarter Sektoren aufgezei [9567]
chnet. Zudem ist der Anzeigebereich etwas
kleiner"
6720 PRINT "als ein voller Sektor."         [2959]
6730 PRINT "Bei der Eingabe von Winkeln (b [7929]
eim Schiessen oder Navigieren) ist zu beac
hten,"
6740 PRINT "dass rechts 0 Grad ist und geg [5221]
en den Uhrzeigersinn gezaehlt wird."
6750 PRINT "PHASENAUFTEILUNG:"              [2437]
6760 PRINT "Da nicht saemtliche befohlenen [8762]
 Aktionen sofort ausgefuehrt werden, ist d
er'
6770 PRINT "folgende Phasenplan von Wichti [4733]
gkeit:"
6780 PRINT "1. Eingabe (!) von Navigation" [3259]
6790 PRINT "   Legen von Minen"             [1781]
6800 PRINT "   Schiessen"                   [957]
6810 WHILE INKEY$=""                            [1095]
6820 WEND                                       [390]
6830 CLS                                        [91]
6840 PRINT "2. Bewegungen werden ausgefueh [3963]
rt (Spieler und Feind)"
6850 PRINT "3. Minen explodieren"           [1311]
6860 PRINT "Wenn der Benutzer eine Demonst [10932]
```

LISTINGS

```
ration der verwendeten Shapes wuenscht, wa
ehle"
6870 PRINT "bei der Eingabe des Schwierigk [5217]
eitsgrades die Option ´eigene"
6880 PRINT "Werte´ und gebe per Hand fuer  [3678]
jeden Typ die"
6890 PRINT "entspr. Anzahlen ein. Dabei er [6190]
scheinen jeweils die Shapes."
6900 PRINT "Auf der Karte erscheinen alle  [6818]
Typen mit ihren Anfangsbuchstaben."
6910 WHILE INKEY$=""                       [1095]
6920 WEND                                  [390]
6930 RETURN                                [555]
6940 ´*********************************** [2371]
**********************************
6950 ´**                              Shap [1106]
es                                  ***
6960 ´*********************************** [2371]
**********************************
6970 RESTORE 7060                          [771]
6980 FOR jj=0 TO 14                        [738]
6990 READ ii                               [356]
7000 MOVE x+ii,y+jj                        [848]
7010 DRAWR -2*ii,0,1,0                     [1086]
7020 MOVE x+ii,y-jj                        [1080]
7030 DRAWR -2*ii,0                         [988]
7040 NEXT                                  [350]
7050 RETURN                                [555]
7060 DATA 14,14,14,14,14,13,13,12,12,11,10 [1876]
,9,8,6,3
7070 RESTORE 7200                          [775]
7080 FOR kk=1 TO 22                        [1144]
7090 READ ii,jj                            [836]
7100 PLOT x+ii,y+jj,1,0                    [1540]
7110 PLOT x-ii,y+jj                        [907]
7120 PLOT x-ii,y-jj                        [947]
7130 PLOT x+ii,y-jj                        [945]
7140 NEXT                                  [350]
7150 MOVE x-10,y                           [772]
7160 DRAW x+10,y                           [490]
7170 MOVE x,y-5                            [682]
7180 DRAW x,y+5                            [160]
7190 RETURN                                [555]
7200 DATA 0,6,1,6,2,6,3,5,4,4,4,5,5,3,5,4, [4039]
6,9,7,8,7,9,8,7,8,8,9,5,9,6,9,7,10,3,10,5,
11,0,11,1,11,2,11,3
7210 FOR ii=0 TO 10                        [800]
7220 MOVE x-6+ii,y+10\(11-ii)              [1671]
7230 DRAW x-6+ii,y-10\(11-ii),1,1          [1975]
7240 NEXT                                  [350]
7250 RETURN                                [555]
7260 FOR ii=5 TO 7 STEP 2                  [1044]
7270 MOVE x-ii,y-ii                        [1825]
7280 DRAW x+ii,y-ii,1,0                    [332]
7290 DRAW x+ii,y+ii                        [139]
7300 DRAW x-ii,y+ii                        [674]
7310 DRAW x-ii,y-ii                        [576]
7320 NEXT                                  [350]
7330 RETURN                                [555]
7340 MOVE x+10,y-6                         [816]
7350 DRAWR -8,0,1,0                        [714]
7360 MOVER 5,0                             [692]
7370 DRAWR 4,4                             [536]
7380 MOVER 3,0                             [700]
7390 DRAWR -20,0                           [365]
7400 MOVER 5,0                             [692]
7410 DRAWR -8,8                            [562]
7420 MOVER -5,0                            [765]
7430 DRAWR 15,0                            [438]
7440 RETURN                                [555]
```

Hubschrauber Simulator





Wollten Sie schon immer einmal wie weiland Stringfellow Hawke im Film "Airwolf" einen Kampfeinsatz mit einem Hubschrauber fliegen?
Dann scheiterte dieses Vorhaben bestimmt zum einen an der Tatsache, daß Hubschrauber ziemlich teuer, und zum anderen daran, daß Kampfeinsätze über Bundesdeutschem Gebiet illegal sind und empfindliche Geldbußen nach sich ziehen können.
Da Sie jedoch diesen Text lesen, kann man davon ausgehen, daß Sie der glückliche Besitzer eines CPC's sind. Ein CPC ist ein Gerät, das so ziemlich alles kann und Ihnen natürlich auch die Möglichkeit bietet, einmal hupend und schraubend in die Lüfte zu steigen.
Alles, was Sie dazu benötigen, ist das nachfolgende Listing und halt Ihren CPC.
Nachdem Sie das Programm gestartet haben, gelangen Sie zunächst einmal ins Hauptmenue, in dem Sie einen Spielmodus wählen können.
Haben Sie sich für einen Level entschieden, so wird das Cockpit des Hubschraubers aufgebaut. Das Cockpit bietet eine Menge Instrumente sowie einen Ausblick auf die Landschaft.
Von dieser Landschaft ist im Moment, außer dem Horizont, noch nichts zu sehen, was sich allerdings im Spielverlauf noch ändern wird.
Unterhalb der Windschutzscheibe befindet sich der Kontrollmonitor, der Ihnen Auskunft über die Wetterlage gibt. Je schlechter das Wetter, um so heftiger die Seitenwinde.
Zum Starten ziehen Sie den Joystick nach hinten und bewegen sich auf FLY LEVEL 4. Nun können Sie mit den Cursortasten bis auf 400 KM/H beschleunigen und Ihren Einsatz erledigen.
Gesteuert wird mit Joystick:
HOCH, RUNTER, RECHTS, LINKS und MG FEUERN.
Mit den Cursortasten:
PFEIL HOCH = Geschwindigkeit erhöhen
PFEIL RUNTER = Geschwindigkeit verringern
COPY-TASTE = Mini-Raketen abfeuern (max. 6 Stck.)
Folgende Instrumente stehen Ihnen zur Verfügung:
NAVI = künstlicher Horizont
TIMER = Flugzeituhr
FUEL = Tankanzeige
MUNI = Munition
FLY LEVEL = Flughöhe
ROT. = Rotorumdrehungen
KM/H = Geschwindigkeit

Tips zum Landen:
Bringen Sie den Hubschrauber in waagerechte Fluglage, senken Sie die Geschwindigkeit auf 0 und gehen Sie auf FLY LEVEL 0.
War die Tankanzeige auf unter 1500 Liter gerutscht, so werden Sie neu betankt.
Durch Drücken von Space und Joystick nach hinten, können Sie die Rotordrehzahl absenken.

Zum erneuten Starten drücken Sie den Joystick nach vorn, bis Sie auf 510 Umdrehungen sind.
GUTEN FLUG!! (Michael Kneip)

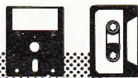
```
1 REM ****************************           [1471]
2 REM *                          *           [403]
3 REM * HUBSCHRAUBERFLUGSIMULATOR *          [3120]
4 REM *                          *           [403]
5 REM *       VERSION   1.0      *           [1504]
6 REM *                          *           [403]
7 REM *       COPYRIGHT  1985    *           [1043]
8 REM *                          *           [403]
9 REM *           by             *           [403]
10 REM *                          *          [403]
11 REM *       Michael  Kneip     *          [1001]
12 REM *                          *          [403]
13 REM ****************************          [1471]
14 GOTO 257                                  [436]
15 BORDER 1                                  [1011]
16 MODE 1:SYMBOL AFTER 140                   [2052]
17 temp=9999                                 [934]
18 muni=10000                                [967]
19 top=6                                     [460]
20 fly=0                                     [297]
21 abzug=10                                  [199]
22 yy=7                                      [330]
23 haus=0                                    [400]
24 hausx=15:rot=500                          [621]
25 farben=5                                  [223]
26 GOSUB 243                                 [934]
27 x=270:y=270:zz=30                         [1185]
28 IF cocpit=2 THEN 29 ELSE 106              [1233]
29 IF start<>2 THEN 133                      [1195]
30 wetter=wetter+1:IF wetter=200 THEN wett   [4327]
er=0:GOTO 143
31 swind=swind+1:IF swind>=seitenwind THEN   [3496]
 swind=0:GOTO 32 ELSE 38
32 wahll=10                                  [652]
33 IF z+1>5 THEN 39 ELSE 34                  [1206]
34 IF z-1<-5 THEN 39 ELSE 35                 [1650]
35 wahlll=INT(RND*20)                        [1198]
36 IF wahlll<10 THEN PLOT 1,1,0:PLOT 0,y:D   [6920]
RAW 640,x:PLOT 0,y+wahll,3:DRAW 640,x-wahl
l:y=y+10:x=x-10:z=z-1
37 IF wahlll>10 THEN PLOT 1,1,0:PLOT 0,y:D   [6794]
RAW 640,x:PLOT 0,y-wahll,3:DRAW 640,x+wahl
l:y=y-10:x=x+10:z=z+1
38 SOUND 1,1000,5,1,0,0,15:SOUND 3,0,5,1,0   [3174]
,0,11
39 LOCATE 1,23:PRINT rot:IF temp<=0 THEN 2   [2359]
08
40 LOCATE 8,1:PRINT x:LOCATE 8,3:PRINT y     [1414]
41 IF kmh>0 AND fly=0 THEN fly=1             [1850]
42 IF kmh=0 AND fly=1 AND INKEY(72)=0 THEN   [1640]
 fly=fly-1
43 IF fly<=0 THEN fly=0                      [629]
44 IF kmh>400 THEN kmh=400                   [1129]
45 IF z=5 THEN 46 ELSE IF INKEY(75)=0 THEN   [3214]
 z=z+1:x=x+10:y=y-10:PLOT 1,1,0:PLOT 0,y+1
0:DRAW 640,x-10
46 IF z=-5 THEN 47 ELSE IF INKEY(74)=0 THE   [8146]
N z=z-1:x=x-10:y=y+10:PLOT 1,1,0:PLOT 0,y-
10:DRAW 640,x+10
47 IF zz=30 THEN 48 ELSE IF INKEY(72)=0 TH   [7010]
EN zz=zz+1:fly=fly-1:x=x+1:y=y+1:PLOT 1,1,
0:PLOT 0,y-1:DRAW 640,x-1
48 IF zz=-30 THEN 49 ELSE IF INKEY(73)=0 T   [5310]
HEN zz=zz-1:fly=fly+1:x=x-1:y=y-1:PLOT 1,1
,0:PLOT 0,y+1:DRAW 640,x+1
49 IF z>=3 AND fly<5 THEN LOCATE 18,17:PRI   [7946]
NT CHR$(24)"DANGER"CHR$(24):LOCATE 18,18:P
RINT CHR$(7)"PULL  ":LOCATE 18,19:PRINT"LE
FT":danger=1
50 IF INKEY(74)=0 THEN LOCATE 17,25:PRINT    [5956]
CHR$(24)CHR$(199)CHR$(24):LOCATE 21,25:PRI
NT" "
51 IF INKEY(75)=0 THEN LOCATE 25,25:PRINT    [5390]
CHR$(24)CHR$(197)CHR$(24):LOCATE 21,25:PRI
NT" "
52 IF INKEY(73)=0 THEN LOCATE 19,25:PRINT    [5285]
CHR$(24)CHR$(198)CHR$(24):LOCATE 21,25:PRI
NT" "
53 IF INKEY(72)=0 THEN LOCATE 23,25:PRINT    [4570]
CHR$(24)CHR$(196)CHR$(24):LOCATE 21,25:PRI
NT" "
54 IF INKEY(74)<>0 AND INKEY(75)<>0 AND IN   [11003]
KEY(73)<>0 AND INKEY(72)<>0 THEN LOCATE 17
,25:PRINT CHR$(199)" "CHR$(198)" ";CHR$(24
);" ";CHR$(24);" "CHR$(196)" ";CHR$(197)
55 IF danger=1 AND z<=2 AND z>=-2 THEN LOC   [7582]
ATE 18,17:PRINT"WETTER":LOCATE 18,18:PRINT
 wetter$:LOCATE 18,19:PRINT"       ":danger
=0
56 IF z<=-3 AND fly<5 THEN LOCATE 18,17:PR   [7096]
INT CHR$(24)"DANGER"CHR$(24):LOCATE 18,18:
PRINT CHR$(7)"PULL  ":LOCATE 18,19:PRINT"R
IGHT":danger=1
57 IF kmh>=100 AND fly<=3 THEN danger=2:LO   [8514]
CATE 18,17:PRINTchr$(24)"DANGER"CHR$(24):L
OCATE 18,18:PRINT CHR$(7)"PULL  ":LOCATE 1
8,19:PRINT"UP   "
58 IF kmh<100 AND danger=2 OR kmh>=100 AND   [10479]
 fly>3 AND danger=2 THEN LOCATE 18,17:PRIN
T"WETTER":LOCATE 18,18:PRINT wetter$:LOCAT
E 18,19:PRINT"       ":danger=3
59 IF recht=1 THEN zahl=zahl+1               [2899]
60 IF zahl=20 THEN zahl=1:GOSUB 191          [2250]
61 IF angrif=2 THEN 62 ELSE 64               [959]
62 IF INKEY(72) THEN GOSUB 198               [1444]
63 IF INKEY(73) THEN GOSUB 203               [1712]
64 IF fly=0 AND temp<1500 AND kmh=0 THEN 2   [2499]
12
65 IF fly=0 AND INKEY(47)=0 AND kmh=0 THEN   [1680]
 215
66 PLOT 1,1,3:PLOT 0,y:DRAW 640,x            [2310]
67 LOCATE 20,23:PRINT fly                    [1767]
68 SOUND 1,1000,5,1,0,0,15:SOUND 3,0,5,1,0   [3174]
,0,11
69 LOCATE 18,8:PRINT CHR$(159):LOCATE 22,8   [2772]
:PRINT CHR$(159)
70 IF muni>0 AND INKEY(76)=0 THEN 159        [1954]
71 LOCATE 1,18:PRINT temp                    [968]
72 temp=temp-11                              [823]
73 IF kmh>9 THEN temp=temp-kmh/10            [1747]
74 IF top>0 AND INKEY(9)=0 THEN 178          [1573]
75 GOSUB 153                                 [924]
76 IF INKEY(0)=0 THEN kmh=kmh+10             [2114]
77 IF INKEY(2)=0 THEN kmh=kmh-10             [979]
78  SOUND 1,1000,5,1,0,0,15:SOUND 3,0,5,1,   [3314]
0,0,11:
79 LOCATE 36,23:PRINT kmh                    [1307]
80 simu=simu-kmh:IF simu<=0 THEN simu=400:   [3283]
GOTO 81 ELSE 101
81 IF farben=5 THEN farben=3 ELSE farben=5   [1468]
82 PEN farben                                [782]
83 IF fly<2 THEN fenster=5                   [1545]
84 IF fly>=2 AND fly<=5 THEN fenster=1       [1868]
85 IF fly=5 THEN fenster=1                   [1500]
86 IF fly>=6 AND fly<=9 THEN fenster=2       [3306]
87 IF fly=9 THEN fenster=2                   [1629]
88 IF fly>=10 AND fly<=13 THEN fenster=3     [2387]
89 IF fly>=14 AND fly<16 THEN fenster=4      [1715]
90 IF fly=14 THEN fenster=4                  [1493]
91 IF fly>16 THEN fenster=5                  [829]
92 IF fenster=1 THEN WINDOW 1,40,12,15       [1879]
93 IF fenster=2 THEN WINDOW 1,40,13,15       [1276]
94 IF fenster=3 THEN WINDOW 1,40,14,15       [1290]
95 IF fenster=4 THEN WINDOW 1,40,15,15       [1104]
96 auswahl=INT(RND*20)                       [1176]
97 IF auswahl<10 AND fenster<5 THEN hausab   [4058]
=INT(RND*40)+1:LOCATE hausab,1:PRINT CHR$(
242)CHR$(243)
98 IF auswahl>10 AND fenster<5 THEN hausab   [8444]
=INT(RND*40)+1:LOCATE hausab,1:PRINT CHR$(
240):LOCATE hausab,2:PRINT CHR$(241)
99 IF fenster<5 THEN LOCATE 1,1:PRINT CHR$   [1705]
(8)
100 WINDOW 1,40,1,25                         [1024]
101 REM                                      [272]
102 REM                                      [272]
103 SOUND 1,1000,5,1,0,0,15:SOUND 3,0,5,1,   [3174]
0,0,11
104 PEN 5                                    [561]
105 GOTO 28                                  [390]
106 PEN 3:LOCATE 2,1:PRINT"NAVI"             [1337]
107 LOCATE 17,22:PRINT"FLY LEVEL"            [2350]
108 LOCATE 2,17:PRINT"FUEL"                  [2000]
109 LOCATE 28,1:PRINT"TIMER"                 [1544]
110 LOCATE 35,17:PRINT"MUNI"                 [1884]
111 LOCATE 2,22:PRINT"ROT."                  [2078]
112 LOCATE 36,22:PRINT"KM/h"                 [1461]
113 PEN 5                                    [561]
114 cocpit=2                                 [480]
115 a=0                                      [324]
116 FOR b=7 TO 14                            [896]
117 a=a+1:IF a=4 THEN a=1                    [1637]
118 PEN a:LOCATE b,20:PRINT CHR$(143):NEXT   [2943]
119 a=3                                      [325]
```

```
120 FOR b=27 TO 34                              [1210]
121 a=a-1:IF a=1 THEN a=3                       [1382]
122 PEN a:LOCATE b,20:PRINT CHR$(143):NEXT      [2943]
123 FOR b=16 TO 25 STEP 3                       [834]
124 a=5                                         [339]
125 PEN a:LOCATE b,1:PRINT CHR$(202):NEXT       [2565]
126 FOR b=19 TO 22 STEP 3                       [1413]
127 a=5                                         [339]
128 PEN a:LOCATE b,3:PRINT CHR$(202):NEXT       [2293]
129 PEN a:LOCATE 17,3:PRINT CHR$(202)           [1616]
130 PEN a:LOCATE 24,3:PRINT CHR$(202)           [1518]
131 GOSUB 226                                   [939]
132 GOTO 28                                     [390]
133 warte=500                                   [927]
134 SOUND 1,1000,5,3,0,0,15:SOUND 3,0,5,3,      [2387]
0,0,11
135 FOR i=1 TO warte:NEXT                       [647]
136 warte=warte-10                              [1272]
137 LOCATE 1,18:PRINT temp                      [968]
138 temp=temp-1                                 [818]
139 IF warte<-50 THEN 140 ELSE 134              [1842]
140 FOR i=1 TO 50:SOUND 1,1000,5,2,0,0,15:      [6771]
SOUND 3,0,5,2,0,0,11:LOCATE 1,18:PRINT tem
p:temp=temp-2:NEXT
141 FOR i=1 TO 50:SOUND 1,1000,5,1,0,0,15:      [6595]
SOUND 3,0,5,1,0,0,11:LOCATE 1,18:PRINT tem
p:temp=temp-2:NEXT
142 LOCATE 18,17:PRINT"WETTER"                  [1313]
143 wahl=INT(RND*4)                             [1578]
144 IF wahl=1 THEN wetter$="GUT   "             [2036]
145 IF wahl=1 THEN seitenwind=10                [1509]
146 IF wahl=2 THEN wetter$="MAESS."             [1122]
147 IF wahl=2 THEN seitenwind=5                 [1733]
148 IF wahl=3 THEN wetter$="SCHLE."             [1730]
149 IF wahl=3 THEN seitenwind=2                 [1698]
150 LOCATE 18,18:PRINT wetter$                  [1139]
151 start=2                                     [634]
152 IF wahl=0 THEN SOUND 1,1000,5,1,0,0,15     [5559]
:SOUND 3,0,5,1,0,0,11::GOTO 143 ELSE GOTO
30
153 sec=sec+1:IF sec=60 THEN sec=0:mi=mi+1      [1946]
154 IF mi=60 THEN mi=0:h=h+1                    [860]
155 IF h=24 THEN h=0                            [1470]
156 LOCATE 28,3:PRINT USING"##:##:##";h;mi      [2679]
;sec
157 SOUND 1,1000,5,1,0,0,15:SOUND 3,0,5,1,      [3174]
0,0,11
158 RETURN                                      [555]
159 FOR p=1 TO 2                                [606]
160 LOCATE 34,18:PRINT muni:muni=muni-2         [1310]
161 SOUND 7,4000,4,7,,,30:LOCATE 18,8:PRIN      [7296]
T CHR$(144):LOCATE 22,8:PRINT CHR$(144):LO
CATE 18,8:PRINT " ":LOCATE 22,8:PRINT " ":

162 LOCATE 8,1:PRINT x:LOCATE 8,3:PRINT y       [1414]
163 IF z=5 THEN 164 ELSE IF INKEY(75)=0 TH      [8061]
EN z=z+1:x=x+10:y=y-10:PLOT 1,1,0:PLOT 0,y
+10:DRAW 640,x-10
164 IF z=-5 THEN 165 ELSE IF INKEY(74)=0 T      [5183]
HEN z=z-1:x=x-10:y=y+10:PLOT 1,1,0:PLOT 0,
y-10:DRAW 640,x+10
165 IF zz=30 THEN 166 ELSE IF INKEY(72)=0       [7386]
THEN zz=zz+1:fly=fly-1:x=x+1:y=y+1:PLOT 1,
1,0:PLOT 0,y-1:DRAW 640,x-1
166 IF zz=-30 THEN 167 ELSE IF INKEY(73)=0      [7155]
 THEN zz=zz+1:fly=fly-1:x=x+1:y=y+1:PLOT 1
,1,0:PLOT 0,y-1:DRAW 640,x-1
167 temp=temp-3                                 [824]
168 LOCATE 1,18:PRINT temp                      [968]
169 PLOT 1,1,3:PLOT 0,y:DRAW 640,x              [2310]
170 LOCATE 20,23:PRINT fly                      [1767]
171 NEXT                                        [350]
172 IF yy=8 THEN LOCATE 18,yy:PRINT CHR$(1      [4482]
54)CHR$(230)CHR$(231)CHR$(230)CHR$(154)
173 IF yy=8 THEN SOUND 130,300,1000,15,1,1      [11443
,15:PLOT 320,276:FOR ra=2 TO 29:FOR k=0 TO
 2*PI STEP PI/5:xxx=ra*SIN(k)+320:yyy=ra*C
OS(k)+276:farbe=INT(RND*8):DRAW xxx,yyy,fa
rbe:NEXT k:NEXT ra
174 IF yy=8 THEN yy=INT(RND*6)+6:angrif=0:      [10838
LOCATE 18,8:PRINT"         ":LOCATE 18,7:
PRINT"          ":LOCATE 18,6:PRINT"
   ":LOCATE 18,9:PRINT"          ":LOCATE 1
8,10:PRINT"          "
175 PLOT 0,0                                    [252]
176 LOCATE 16,8:PRINT"             "            [1428]
177 GOTO 28                                     [390]
178 LOCATE 18,8:PRINT CHR$(238):LOCATE 22,      [3409]
8:PRINT CHR$(238)
179 INK 0,24:FOR i=1 TO 50:NEXT:INK 0,0         [1019]
180 FOR n=100 TO 800 STEP 15                    [1266]
181 SOUND 1,n,2,15                              [1455]
182 NEXT                                        [350]
183 FOR xx=1 TO 7 STEP 0.5                      [1689]
184 SOUND 1,400,5,xx,,,1                        [1526]
185 NEXT                                        [350]
186 SOUND 1,0,50                                [1112]
187 top=top-2                                   [1036]
188 LOCATE 18,8:PRINT " ":LOCATE 22,8:PRIN      [1810]
T " "
189 IF yy=8 THEN 172                            [1170]
190 GOTO 74                                     [392]
191 FOR n=500 TO 100 STEP -15                   [1692]
192 SOUND 1,n,4                                 [1145]
193 NEXT                                        [350]
194 SOUND 1,0,30                                [1148]
195 IF angrif=2 AND bewaf=1 THEN 235            [753]
196 angrif=2                                    [940]
197 RETURN                                      [555]
198 auf=auf+1                                   [931]
199 IF auf=8 THEN auf=0:IF yy<15 THEN yy=y      [2080]
y+1
200 LOCATE 18,yy:PRINT CHR$(154)CHR$(230)C      [2416]
HR$(231)CHR$(230)CHR$(154)
201 IF yy-1>5 THEN LOCATE 17,yy-1:PRINT "       [2566]
    "
202 RETURN                                      [555]
203 auf=auf+1                                   [931]
204 IF auf=8 THEN auf=0:IF yy>6 THEN yy=yy      [2189]
-1
205 LOCATE 18,yy:PRINT CHR$(154)CHR$(230)C      [2416]
HR$(231)CHR$(230)CHR$(154)
206 IF yy<15 THEN LOCATE 17,yy+1:PRINT "        [2367]
    "
207 RETURN                                      [555]
208 INK 0,0:FOR i=1 TO 5000:NEXT:CLS:LOCAT      [5444]
E 13,5:PRINT"G A M E   O V E R":FOR i=1 TO
 5000:NEXT
209 MODE 2:INK 0,0                              [1473]
210 INPUT"Noch einmal";wahl$                    [1890]
211 IF wahl$="j" THEN RUN ELSE CLS:END          [2291]
212 MODE 2                                      [513]
213 PRINT "Ihr Hubschrauber ist wieder fri      [5875]
sch aufgetank und mit 6 Raketen versorgt"
214 FOR i=1 TO 5000:NEXT:CLEAR:PLOT 1,1,1:      [3065]
RUN
215 SOUND 1,1000,5,3,0,0,15:SOUND 3,0,5,3,      [2387]
0,0,11
216 warte=0                                     [924]
217 IF INKEY(73)=0 THEN warte=warte+10:rot      [3236]
=rot-10
218 IF INKEY(72)=0 THEN warte=warte-10:rot      [1358]
=rot+10
219 IF warte>500 THEN 217 ELSE IF warte<-5      [1986]
0 THEN 28 ELSE 220
220 FOR I=1 TO WARTE:NEXT:SOUND 1,1000,5,3      [2619]
,0,0,15:SOUND 3,0,5,3,0,0,11
221 LOCATE 1,18:PRINT temp                      [968]
222 temp=temp-2                                 [825]
223 LOCATE 1,23:PRINT rot:IF temp<=0 THEN       [2303]
256
224 IF rot<0 THEN rot=0 ELSE IF rot>500 TH      [2745]
EN rot=500
225 GOTO 217                                    [428]
226 SYMBOL AFTER 154                            [1454]
227 SYMBOL 154,&X0,&X0,&X0,&X0,&X0,&X11111      [2919]
111,&X0
228 SYMBOL 230,&X0,&X0,&X0,&X0,&X0,&X11111      [3692]
111,&X100100,&X1011010
229 SYMBOL 231,&X0,&X11000,&X111100,&X1011      [4410]
010,&X1011010,&X11111111,&X111100,&X11000
230 SYMBOL 240,&X11000,&X111100,&X1111110,      [5336]
&X10101011,&X1011101,&X1010110,&X110100,&X
11000
231 SYMBOL 241,&X1011010,&X111100,&X11000,      [4066]
&X11000,&X11000,&X11000,&X11100,&X11010
232 SYMBOL 242,&X11111,&X100000,&X1000000,      [4328]
&X11111111,&X10101011,&X10101011,&X1111111
1
233 SYMBOL 243,&X11100000,&X1100000,&X1010      [4732]
0000,&X1100000,&X10100000,&X1000000,&X1000
0000
234 RETURN                                      [555]
235 FOR xx=1 TO 7 STEP 0.5:SOUND 1,400,5,x      [3928]
x,,,1:NEXT:SOUND 1,0,50
236 PLOT 0, 90, 5:PLOT 0, 90:PLOT 0, 90:PL      [11043]
OT 0, 90:PLOT 0, 90:PLOT 0, 90:PLOT 0, 90:
PLOT 0, 90:PLOT 640, 160:DRAW 640, 330:DRA
```

```
W 0, 330:DRAW 0, 160:PLOT 0, 160:DRAW 640,
 160:PLOT 380, 240:DRAW 430, 330:PLOT 380,
 240
237 DRAW 450, 270:PLOT 380, 240:DRAW 420,  [10229]
190:PLOT 380, 240:DRAW 430, 230:PLOT 380,
240:DRAW 370, 180:PLOT 380, 240:DRAW 330,
200:PLOT 380, 240:DRAW 310, 310:PLOT 380,
240:DRAW 360, 320:PLOT 380, 240:DRAW 300,
250
238 PLOT 0, 310:DRAW 640, 290:PLOT 0, 270:  [8237]
DRAW 640, 280:DRAW 640, 260:DRAW 0, 290:PL
OT 0, 240:DRAW 640, 310:PLOT 640, 240:DRAW
-10, 260:PLOT 0, 210:DRAW 640, 250:PLOT 0,
 190:DRAW 640, 210:PLOT 640, 190:DRAW 230,
 160
239 PLOT 40, 160:DRAW 60, 330:PLOT 90, 330  [10095]
:DRAW 90, 160:PLOT 130, 330:DRAW 70, 160:P
LOT 120, 160:DRAW 140, 330:PLOT 150, 160:P
LOT 150, 330:DRAW 160, 160:DRAW 190, 160:D
RAW 190, 330:DRAW 210, 160:DRAW 230, 330
240 DRAW 260, 160:DRAW 310, 160:DRAW 300,  [10876]
330:DRAW 350, 330:DRAW 370, 160:DRAW 410,
160:DRAW 440, 330:DRAW 480, 160:DRAW 530,
330:DRAW 560, 330:PLOT 540, 160:DRAW 490,
330:PLOT 560, 330:DRAW 590, 160:DRAW 610,
160
241 DRAW 630, 330:DRAW 580, 330:DRAW 560,  [922]
160
242 GOTO 208                                [421]
243 PLOT 0, 390:DRAW 50, 340:PLOT 640, 390  [10617]
:PLOT 590, 340:DRAW 590, 340:PLOT 590, 340
:PLOT 640, 390:DRAW 590, 340:DRAW 50, 340:
PLOT 0, 150:DRAW 100, 150:DRAW 140, 110:PL
OT 640, 150:PLOT 640, 150:DRAW 540, 150
244 DRAW 500, 110:DRAW 420, 110:DRAW 380,  [9662]
70:PLOT 140, 110:DRAW 220, 110:DRAW 260, 7
0:DRAW 380, 70:PLOT 640, 100:DRAW 590, 100
:DRAW 550, 60:DRAW 470, 60:DRAW 430, 20:DR
AW 450, 0:PLOT 0, 100:DRAW 50, 100:DRAW 90
, 60
245 DRAW 130, 60:DRAW 140, 60:DRAW 150, 60  [12408]
:DRAW 170, 60:DRAW 210, 20:DRAW 190, 0:PLO
T 100, 150:PLOT 0, 155:DRAW 105, 155:DRAW
145, 115:DRAW 225, 115:DRAW 265, 75:DRAW 3
75, 75:PLOT 375, 75:PLOT 375, 75:DRAW 415,
 115
246 DRAW 495, 115:DRAW 535, 155:DRAW 640,  [9947]
155:PLOT 640, 155:DRAW 105, 155:PLOT 0, 34
0:DRAW 50, 340:DRAW 640, 340:PLOT 640, 335
:DRAW 0, 335:PLOT 45, 340:DRAW 0, 385:PLOT
 600, 340:PLOT 640, 385:DRAW 595, 340
247 PLOT 0, 390:DRAW 50, 340:PLOT 640, 390  [10617]
:PLOT 590, 340:DRAW 590, 340:PLOT 590, 340
:PLOT 640, 390:DRAW 590, 340:DRAW 50, 340:
PLOT 0, 150:DRAW 100, 150:DRAW 140, 110:PL
OT 640, 150:PLOT 640, 150:DRAW 540, 150
248 DRAW 500, 110:DRAW 420, 110:DRAW 380,  [9662]
70:PLOT 140, 110:DRAW 220, 110:DRAW 260, 7
0:DRAW 380, 70:PLOT 640, 100:DRAW 590, 100
:DRAW 550, 60:DRAW 470, 60:DRAW 430, 20:DR
AW 450, 0:PLOT 0, 100:DRAW 50, 100:DRAW 90
, 60
249 DRAW 130, 60:DRAW 140, 60:DRAW 150, 60  [12408]
:DRAW 170, 60:DRAW 210, 20:DRAW 190, 0:PLO
T 100, 150:PLOT 0, 155:DRAW 105, 155:DRAW
145, 115:DRAW 225, 115:DRAW 265, 75:DRAW 3
75, 75:PLOT 375, 75:PLOT 375, 75:DRAW 415,
 115
250 DRAW 495, 115:DRAW 535, 155:DRAW 640,  [9402]
155:PLOT 640, 155:DRAW 105, 155:PLOT 0, 34
0:DRAW 50, 340:DRAW 640, 340:PLOT 640, 335
:DRAW 0, 335:PLOT 45, 340:DRAW 0, 385:PLOT
 600, 340:PLOT 1,1,3:PLOT 640, 385:DRAW 59
5, 340
251 PLOT 100, 150:PLOT 620, 370:DRAW 20, 3  [8910]
70:PLOT 320, 340:DRAW 320, 400:PLOT 370, 4
00:DRAW 360, 340:PLOT 270, 400:PLOT 280, 3
40:DRAW 270, 400:PLOT 220, 400:DRAW 240, 3
40:PLOT 400, 340:PLOT 360, 340:DRAW 390, 3
40
252 PLOT 420, 400:DRAW 400, 340:PLOT 320,  [11296]
150:DRAW 270, 150:DRAW 370, 150:DRAW 370,
80:DRAW 270, 80:DRAW 270, 150:PLOT 370, 15
0:DRAW 380, 140:DRAW 380, 90:DRAW 370, 80:
DRAW 270, 80:DRAW 260, 90:DRAW 260, 140
253 DRAW 270, 150:PLOT 0, 110:DRAW 60, 110  [8114]
:DRAW 100, 70:DRAW 180, 70:DRAW 230, 20:PL
OT 410, 20:DRAW 460, 70:DRAW 540, 70:DRAW
580, 110:DRAW 640, 110:PLOT 230, 20:DRAW 2
30, 0:PLOT 410, 0:DRAW 410, 20
254 PLOT 1,1,5:PLOT 0,340:DRAW 640,340      [1936]
255 RETURN                                  [555]
256 REM                                     [272]
257 MODE 2                                  [513]
258 PRINT"[1]......Mit Gegner(mit  Raketen  [2581]
)
259 PRINT"[2]......Mit Gegner(ohne Raketen  [2516]
)
260 PRINT"[3]......Ohne Gegner"             [1777]
261 PRINT"[4]......Spiel beenden"           [2465]
262 PRINT"[5]......Spiel abspeichern"       [2875]
263 PRINT:PRINT"Ihre Wahl"                  [1523]
264 a$=INKEY$:IF a$=""THEN 264 ELSE 265     [1506]
265 IF a$="1" THEN bewaf=1:recht=1          [1845]
266 IF a$="2" THEN bewaf=0:recht=1          [952]
267 IF a$="3" THEN recht=0:recht=0          [2388]
268 IF a$="4" THEN MODE 1:CLS:END           [1870]
269 IF a$="5" THEN 271                      [308]
270 GOTO 15                                 [391]
271 REM save                                [508]
272 MODE 2:CLS                              [1257]
273 PRINT"[1] = 1000 Baud"                  [1730]
274 PRINT"[2] = 2000 Baud"                  [1851]
275 PRINT"[3] = 3500 Baud"                  [1634]
276 PRINT"[4] = Spiel starten"              [2708]
277 PRINT                                   [361]
278 PRINT"Ihre Wahl [1-4] ?"                [1937]
279 a$=INKEY$:IF a$="" THEN 279 ELSE 280    [1845]
280 IF a$="1" THEN SPEED WRITE 0:GOTO 285   [2996]
281 IF a$="2" THEN SPEED WRITE 1:GOTO 285   [2304]
282 IF a$="3" THEN POKE &B8D1,2:POKE &B8D2  [2619]
,23:GOTO 285
283 IF a$="4" THEN RUN                      [908]
284 GOTO 273                                [412]
285 CLS:SAVE"hubschraubers."                [1248]
286 GOTO 273                                [412]
```

Gangster



Vor etwa drei Wochen wurden in London mehrere Banken überfallen. Die 20 Mitglieder der Bande, auf deren Konto die Überfälle gehen, haben die Beute an verschiedenen Stellen in der Stadt "abgebunkert", um die Polizei in die Irre zu führen.
Der Boß der Gang ist nun allerdings der Meinung, daß genug Gras über die Sache gewachsen ist und die "Kohle" abgeholt werden kann.
Die Bandenmitglieder werden also in einem Firmenwagen auf den Weg geschickt und sollen zumindest einen Teil der Beute wieder einsammeln. Allerdings hat sich der Boß ein wenig verkalkuliert, die Polizei hat von der Aktion Wind bekommen und beschattet das Auto.
Überall, entlang der bereits gefahrenen Strecke, werden Polizisten postiert, die den Auftrag haben, einen aus der Gang "hopp" zu nehmen, wenn er das nächste Mal vorbeikommt.
Passen Sie also auf, daß Sie keine Strecke zweimal fahren und bauen Sie auch keine Unfälle, indem Sie vor Häuser fahren.
Eine Hilfe ist, daß Sie auf dem Weg durch die Stadt noch ein paar arbeitslose Diebe einstellen können, die Ihnen bei Ihrem Job helfen.
Das Programm ist vollständig in M-Code geschrieben und somit ziemlich schnell. Eine Kurzanleitung und eine High Score-Tabelle ist integriert.

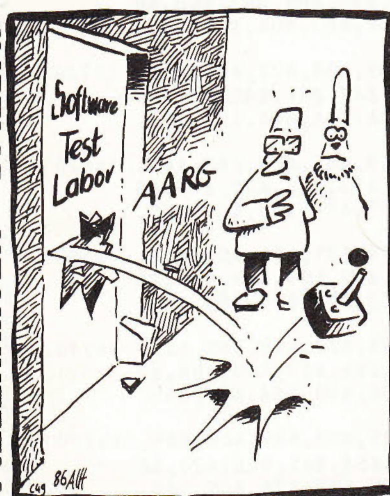
(Thorsten Kuhn)

```
10 ' ******************************         [1252]
20 ' **** (c) by Thorsten Kuhn ****         [988]
30 ' ******************************         [1252]
40 '                                        [117]
50 '****** BASIC LADER ******               [1316]
```

```
60 '                                                  [117]
70 dat=0:sum=0:zeile=210                              [1153]
80 FOR adr=&581A TO &65CD                             [1203]
90 READ byte:dat=dat+1                                [956]
100 sum=sum+byte                                      [874]
110 POKE adr,byte                                     [84]
120 IF dat < 30 AND adr < &65CD THEN 160              [1536]
130 READ chksum                                       [1222]
140 IF chksum <> sum THEN PRINT "Fehler in            [4252]
 Zeile : ";zeile
150 zeile=zeile+10:sum=0:dat=0                        [1738]
160 NEXT adr                                          [547]
170 CALL &581A                                        [554]
180 '                                                 [117]
190 '******  MC-DATAS  ******                          [1114]
200 '                                                 [117]
210 DATA &CD,&FF,&BB,&CD,&BA,&BB,&CD,&4E,&  [3794]
BB,&CD,&00,&BB,&CD,&48,&BB,&11,&FB,&00,&21
,&FD,&66,&CD,&AB,&BB,&3E,&FB,&21,&9C,&5E,&
CD, 4571
220 DATA &A8,&BB,&3E,&FD,&21,&A4,&5E,&CD,&  [6353]
A8,&BB,&3E,&FC,&21,&AC,&5E,&CD,&A8,&BB,&3E
,&00,&06,&03,&0E,&03,&CD,&32,&BC,&3E,&01,&
06, 3294
230 DATA &1A,&0E,&1A,&CD,&32,&BC,&3E,&02,&  [6207]
06,&18,&0E,&18,&CD,&32,&BC,&3E,&03,&06,&0B
,&0E,&0B,&CD,&32,&BC,&01,&09,&09,&CD,&38,&
BC, 2102
240 DATA &21,&CE,&65,&06,&14,&36,&0A,&23,&  [4716]
36,&00,&23,&10,&F8,&21,&E2,&65,&06,&FA,&36
,&9A,&23,&10,&FB,&21,&E2,&65,&06,&0A,&36,&
30, 2422
250 DATA &23,&36,&30,&23,&36,&31,&23,&36,&  [6125]
30,&23,&36,&20,&11,&15,&00,&19,&10,&EC,&CD
,&A7,&BC,&3E,&01,&CD,&0E,&BC,&21,&0A,&03,&
22, 1958
260 DATA &3B,&67,&11,&12,&24,&ED,&53,&3D,&  [6071]
67,&3E,&01,&32,&3F,&67,&2A.&3B,&67,&ED,&5B
,&3D,&67,&3A,&3F,&67,&CD,&2C,&BC,&CD,&44,&
BC, 2814
270 DATA &2A,&3B,&67,&ED,&5B,&3D,&67,&3A,&  [6796]
3F,&67,&24,&15,&2C,&1D,&3C,&22,&3B,&67,&ED
,&53,&3D,&67,&32,&3F,&67,&FE,&05,&C2,&BE,&
58, 2737
280 DATA &21,&DC,&5E,&7E,&FE,&FF,&CA,&FC,&  [6200]
58,&CD,&5A,&BB,&23,&C3,&EF,&58,&CD,&03,&BB
,&CD,&06,&BB,&FE,&45,&C8,&FE,&65,&C8,&FE,&
54, 4772
290 DATA &CA,&DE,&5D,&FE,&74,&CA,&DE,&5D,&  [6984]
FE,&53,&CA,&3F,&59,&FE,&73,&CA,&3F,&59,&FE
,&41,&CA,&29,&59,&FE,&61,&CA,&29,&59,&C3,&
FC, 4599
300 DATA &58,&21,&91,&5F,&7E,&FE,&FF,&CA,&  [6883]
39,&59,&CD,&5A,&BB,&23,&C3,&2C,&59,&CD,&06
,&BB,&C3,&A4,&58,&21,&8C,&64,&7E,&FE,&FF,&
CA, 4144
310 DATA &4F,&59,&CD,&5A,&BB,&23,&C3,&42,&  [5033]
59,&CD,&03,&BB,&CD,&06,&BB,&FE,&31,&CA,&6C
,&59,&FE,&32,&CA,&7B,&59,&FE,&33,&CA,&8A,&
59, 3982
320 DATA &FE,&34,&CA,&99,&59,&C3,&4F,&59,&  [4990]
21,&01,&40,&22,&2C,&67,&21,&0A,&00,&22,&27
,&67,&C3,&A5,&59,&21,&01,&30,&22,&2C,&67,&
21, 2356
330 DATA &32,&00,&22,&27,&67,&C3,&A5,&59,&  [5770]
21,&01,&20,&22,&2C,&67,&21,&5A,&00,&22,&27
,&67,&C3,&A5,&59,&21,&01,&10,&22,&2C,&67,&
21, 1934
340 DATA &82,&00,&22,&27,&67,&3E,&14,&32,&  [5719]
F3,&66,&21,&00,&00,&22,&F7,&66,&21,&00,&00
,&22,&F5,&66,&21,&02,&02,&22,&F9,&66,&21,&
6B, 2175
350 DATA &10,&22,&FB,&66,&3E,&FB,&32,&25,&  [5080]
67,&2A,&F7,&66,&11,&2A,&00,&19,&22,&F7,&66
,&3E,&20,&32,&29,&67,&21,&19,&01,&22,&E3,&
66, 2474
360 DATA &3E,&01,&32,&E5,&66,&CD,&14,&BC,&  [6373]
21,&01,&01,&11,&B4,&5E,&06,&27,&E5,&CD,&75
,&BB,&E1,&1A,&CD,&5A,&BB,&24,&13,&10,&F3,&
3E, 3075
370 DATA &0F,&CD,&5A,&BB,&3E,&02,&CD,&5A,&  [7145]
BB,&C3,&10,&5A,&3E,&0F,&CD,&5A,&BB,&3E,&03
,&CD,&5A,&BB,&CD,&06,&B9,&06,&28,&2A,&FB,&
66, 3287
380 DATA &23,&22,&FB,&66,&7E,&57,&3A,&29,&  [6335]
67,&BA,&DA,&31,&5A,&2A,&F9,&66,&CD,&75,&BB
,&3A,&25,&67,&CD,&5A,&BB,&2A,&F9,&66,&24,&
22, 3425
390 DATA &F9,&66,&10,&DB,&2A,&F9,&66,&26,&  [7184]
01,&2C,&22,&F9,&66,&7C,&85,&FE,&1A,&C2,&10
,&5A,&CD,&09,&B9,&3A,&25,&67,&2E,&A3,&BD,&
CA, 3481
400 DATA &6F,&5A,&3E,&01,&32,&29,&67,&3E,&  [6468]
A3,&32,&25,&67,&21,&02,&01,&22,&F9,&66,&21
,&29,&00,&22,&FB,&66,&C3,&06,&5A,&21,&15,&
24, 2136
410 DATA &CD,&75,&BB,&3E,&FC,&CD,&5A,&BB,&  [5412]
21,&0D,&03,&CD,&75,&BB,&3E,&FC,&CD,&5A,&BB
,&C3,&88,&5A,&C3,&35,&5C,&3E,&0F,&CD,&5A,&
BB, 3979
420 DATA &3E,&01,&CD,&5A,&BB,&ED,&4B,&2C,&  [6735]
67,&19,&ED,&52,&ED,&5A,&ED,&52,&ED,&5A,&ED
,&52,&0B,&3E,&00,&B8,&C2,&99,&5A,&B9,&C2,&
99, 3866
430 DATA &5A,&2A,&E3,&66,&CD,&75,&BB,&3E,&  [6322]
90,&CD,&5A,&BB,&CD,&24,&BB,&FE,&01,&CA,&0A
,&5B,&FE,&02,&CA,&27,&5B,&FE,&04,&CA,&44,&
5B, 3851
440 DATA &FE,&08,&CA,&61,&5B,&FE,&10,&CA,&  [5578]
7E,&5B,&CD,&1B,&BB,&D2,&04,&5B,&3E,&00,&CD
,&1E,&BB,&C2,&0A,&5B,&3E,&02,&CD,&1E,&BB,&
C2, 3524
450 DATA &27,&5B,&3E,&08,&CD,&1E,&BB,&C2,&  [6713]
44,&5B,&3E,&01,&CD,&1E,&BB,&C2,&61,&5B,&3E
,&09,&CD,&1E,&BB,&C2,&7E,&5B,&3A,&E5,&66,&
C3, 3330
460 DATA &BD,&5A,&3E,&01,&32,&E5,&66,&2A,&  [6922]
E3,&66,&2D,&22,&E3,&66,&3E,&01,&BD,&C2,&86
,&5B,&2A,&E3,&66,&2E,&19,&22,&E3,&66,&C3,&
86, 3307
470 DATA &5B,&3E,&02,&32,&E5,&66,&2A,&E3,&  [7035]
66,&2C,&22,&E3,&66,&3E,&1A,&BD,&C2,&86,&5B
,&2A,&E3,&66,&2E,&02,&22,&E3,&66,&C3,&86,&
5B, 3212
480 DATA &3E,&04,&32,&E5,&66,&2A,&E3,&66,&  [6006]
25,&22,&E3,&66,&3E,&00,&BC,&C2,&86,&5B,&2A
,&E3,&66,&26,&28,&22,&E3,&66,&C3,&86,&5B,&
3E, 3181
490 DATA &08,&32,&E5,&66,&2A,&E3,&66,&24,&  [5875]
22,&E3,&66,&3E,&29,&BC,&C2,&86,&5B,&2A,&E3
,&66,&26,&01,&22,&E3,&66,&C3,&86,&5B,&3E,&
10, 3140
500 DATA &32,&E5,&66,&C3,&BA,&5A,&2A,&E3,&  [6228]
66,&25,&2D,&CD,&1A,&BC,&11,&00,&08,&19,&19
,&19,&7E,&32,&3A,&67,&2A,&E3,&66,&CD,&75,&
BB, 3041
510 DATA &3E,&FD,&CD,&5A,&BB,&3A,&3A,&67,&  [6110]
FE,&10,&CA,&CD,&5B,&FE,&08,&CA,&DF,&5B,&FE
,&FF,&CA,&F8,&5B,&FE,&33,&CA,&BD,&5B,&C3,&
88, 4730
520 DATA &5A,&21,&1F,&64,&CD,&AA,&BC,&2A,&  [5134]
F3,&66,&23,&22,&F3,&66,&C3,&88,&5A,&CD,&A7
,&BC,&21,&28,&64,&CD,&AA,&BC,&21,&31,&64,&
CD, 3722
530 DATA &AA,&BC,&C3,&E5,&5B,&21,&0D,&64,&  [6996]
CD,&AA,&BC,&2A,&F3,&66,&2B,&22,&F3,&66,&11
,&00,&00,&AF,&ED,&52,&CA,&1B,&5D,&C3,&88,&
5A, 3645
540 DATA &2A,&F5,&66,&23,&22,&F5,&66,&21,&  [5543]
16,&64,&CD,&AA,&BC,&2A,&F5,&66,&AF,&ED,&5B
,&F7,&66,&ED,&52,&C2,&88,&5A,&2A,&2C,&67,&
11, 3709
550 DATA &01,&00,&AF,&ED,&52,&CA,&B6,&59,&  [5179]
2A,&2C,&67,&25,&25,&25,&25,&22,&2C,&67,&2A
,&27,&67,&11,&0A,&00,&19,&22,&27,&67,&C3,&
B6, 2275
560 DATA &59,&3E,&01,&32,&E2,&66,&21,&E9,&  [6373]
66,&22,&E7,&66,&3A,&E2,&66,&FE,&01,&CA,&5A
,&5C,&FE,&02,&CA,&69,&5C,&FE,&03,&CA,&78,&
5C, 3616
570 DATA &FE,&04,&CA,&98,&5C,&C3,&88,&5A,&  [5331]
21,&01,&0A,&22,&38,&67,&2A,&F3,&66,&22,&30
,&67,&C3,&9B,&5C,&21,&01,&19,&22,&38,&67,&
2A, 2670
580 DATA &F5,&66,&22,&30,&67,&C3,&9B,&5C,&  [6309]
21,&EE,&66,&06,&04,&ED,&5B,&E7,&66,&1A,&77
,&13,&23,&10,&FA,&36,&20,&21,&01,&25,&22,&
38, 2730
590 DATA &67,&2A,&27,&67,&22,&30,&67,&C3,&  [5689]
9B,&5C,&C3,&8B,&5A,&21,&E8,&03,&22,&2A,&67
,&21,&00,&00,&22,&2E,&67,&06,&00,&3A,&2E,&
67, 2214
600 DATA &3C,&32,&2E,&67,&ED,&5B,&2A,&67,&  [7304]
2A,&30,&67,&AF,&ED,&52,&DA,&C4,&5C,&04,&22
,&30,&67,&C3,&B0,&5C,&2A,&38,&67,&CD,&75,&
BB, 3287
610 DATA &3E,&30,&80,&2A,&E7,&66,&77,&21,&  [5458]
E7,&66,&34,&CD,&5A,&BB,&2A,&38,&67,&24,&22
```

```
,&38,&67,&3A,&2E,&67,&FE,&01,&CA,&F6,&5C,&
FE, 3318
620 DATA &02,&CA,&FF,&5C,&FE,&03,&CA,&08,& [5017]
5D,&FE,&04,&CA,&11,&5D,&21,&64,&00,&22,&2A
,&67,&C3,&A7,&5C,&21,&0A,&00,&22,&2A,&67,&
C3, 2864
630 DATA &A7,&5C,&21,&01,&00,&22,&2A,&67,& [5709]
C3,&A7,&5C,&3A,&E2,&66,&3C,&32,&E2,&66,&C3
,&3A,&5C,&3E,&00,&32,&DC,&66,&21,&CE,&65,&
E5, 3098
640 DATA &3A,&DC,&66,&3C,&32,&DC,&66,&5E,& [5412]
23,&56,&2A,&F5,&66,&AF,&ED,&52,&DA,&58,&5E
,&3A,&DC,&66,&CB,&27,&67,&3E,&14,&94,&CA,&
52, 3549
650 DATA &5D,&32,&DD,&66,&ED,&4B,&DD,&66,& [7659]
11,&E1,&65,&21,&DF,&65,&ED,&B8,&ED,&5B,&F5
,&66,&E1,&73,&23,&72,&21,&E2,&65,&3A,&DC,&
66, 4126
660 DATA &47,&11,&19,&00,&19,&10,&FA,&AF,& [6646]
11,&19,&00,&ED,&52,&22,&E0,&66,&11,&E2,&65
,&AF,&ED,&52,&AF,&EB,&21,&E1,&00,&ED,&52,&
22, 3159
670 DATA &DD,&66,&ED,&4B,&DD,&66,&03,&11,& [6202]
DB,&66,&21,&C2,&66,&ED,&B8,&21,&73,&61,&7E
,&CD,&5A,&BB,&23,&7E,&FE,&FF,&C2,&90,&5D,&
21, 4036
680 DATA &16,&0A,&CD,&75,&BB,&2A,&E0,&66,& [6612]
06,&05,&11,&EE,&66,&1A,&77,&23,&13,&10,&FA
,&3E,&0F,&CD,&5A,&BB,&3E,&02,&CD,&5A,&BB,&
06, 2853
690 DATA &14,&CD,&06,&BB,&FE,&7F,&CA,&7F,& [5246]
5E,&FE,&0D,&CA,&94,&5E,&FE,&20,&DA,&BB,&5D
,&CD,&5A,&BB,&77,&23,&10,&E7,&3E,&0F,&CD,&
5A, 3972
700 DATA &BB,&3E,&01,&CD,&5A,&BB,&3E,&0C,& [6263]
CD,&5A,&BB,&21,&B4,&63,&7E,&CD,&5A,&BB,&23
,&7E,&FE,&FF,&C2,&E6,&5D,&21,&09,&08,&CD,&
75, 3767
710 DATA &BB,&21,&E2,&65,&3E,&00,&32,&DF,& [5720]
66,&E5,&3E,&09,&CD,&6F,&BB,&E1,&06,&19,&7E
,&CD,&5A,&BB,&23,&10,&F9,&3E,&0A,&CD,&5A,&
BB, 3505
720 DATA &3E,&0D,&CD,&5A,&BB,&3A,&DF,&66,& [4982]
3C,&32,&DF,&66,&FE,&0A,&C2,&FF,&5D,&3E,&0B
,&CD,&5A,&BB,&21,&7D,&65,&7E,&FE,&FF,&CA,&
3A, 3890
730 DATA &5E,&CD,&5A,&BB,&23,&C3,&2D,&5E,& [5758]
21,&3A,&64,&E5,&CD,&AA,&BC,&E1,&D2,&3D,&5E
,&11,&09,&00,&19,&7E,&FE,&FF,&CA,&52,&5E,&
C3, 3777
740 DATA &3D,&5E,&CD,&06,&BB,&C3,&A4,&58,& [5988]
E1,&23,&23,&3A,&DC,&66,&FE,&0A,&C2,&23,&5D
,&CD,&14,&BC,&21,&8A,&62,&7E,&CD,&5A,&BB,&
23, 3586
750 DATA &7E,&FE,&FF,&C2,&69,&5E,&CD,&06,& [6401]
BB,&FE,&20,&DA,&74,&5E,&C3,&DE,&5D,&3E,&14
,&B8,&CA,&BB,&5D,&3E,&08,&CD,&5A,&BB,&3E,&
10, 4028
760 DATA &CD,&5A,&BB,&04,&2B,&C3,&BB,&5D,& [6429]
36,&20,&23,&10,&FB,&C3,&DE,&5D,&10,&38,&7C
,&82,&AA,&82,&AA,&8A,&00,&00,&38,&78,&FF,&
FF, 3522
770 DATA &66,&66,&38,&38,&12,&3C,&50,&10,& [5356]
28,&6C,&47,&61,&6E,&67,&73,&74,&65,&72,&3A
,&20,&20,&20,&20,&20,&20,&20,&20,&20,&47,&
65, 1983
780 DATA &6C,&64,&3A,&A3,&20,&20,&20,&20,& [5397]
20,&20,&20,&6B,&6D,&2F,&68,&3A,&20,&20,&20
,&20,&0A,&0A,&09,&09,&09,&09,&09,&09,&09,&
43, 1356
790 DATA &6F,&70,&79,&72,&69,&67,&68,&74,& [5756]
20,&62,&79,&20,&54,&68,&6F,&72,&73,&74,&65
,&6E,&20,&4B,&75,&68,&6E,&0A,&0A,&0D,&09,&
09, 2513
800 DATA &09,&09,&09,&09,&09,&09,&09,&09,& [5171]
09,&09,&09,&09,&09,&09,&09,&A4,&20,&31,&39
,&38,&36,&0F,&03,&0A,&0A,&0A,&0A,&0A,&0A,&
0A, 635
810 DATA &0A,&0A,&0A,&0D,&09,&09,&09,&09,& [5018]
09,&09,&09,&09,&09,&09,&09,&09,&09,&09,&09
,&09,&47,&41,&4E,&47,&53,&54,&45,&52,&0F,&
01, 806
820 DATA &0F,&02,&0A,&0A,&0A,&0A,&0A,&0A,& [6710]
0A,&0D,&09,&09,&09,&09,&09,&09,&09,&09,&09
,&09,&09,&3C,&41,&3E,&20,&53,&70,&69,&65,&
6C, 927
830 DATA &61,&6E,&6C,&65,&69,&74,&75,&6E,& [6655]
67,&0A,&0A,&0D,&09,&09,&09,&09,&3C,&53,&3E
,&20,&53,&74,&61,&72,&74,&20,&20,&3C,&54,&
3E, 2069
840 DATA &20,&54,&6F,&70,&2D,&54,&65,&6E,& [6693]
20,&20,&3C,&45,&3E,&20,&45,&6E,&64,&65,&0F
,&01,&FF,&0C,&0A,&0A,&0D,&09,&09,&09,&09,&
09, 1707
850 DATA &09,&09,&09,&09,&09,&09,&09,&09,& [6080]
0F,&03,&53,&70,&69,&65,&6C,&61,&6E,&6C,&65
,&69,&74,&75,&6E,&67,&0F,&01,&0A,&0A,&0A,&
0D, 1625
860 DATA &09,&2D,&53,&70,&69,&65,&6C,&66,& [5373]
69,&67,&75,&72,&65,&6E,&0A,&0A,&0D,&09,&09
,&0F,&02,&FB,&0F,&01,&20,&48,&61,&75,&73,&
0A, 2093
870 DATA &0D,&09,&09,&0F,&03,&A3,&0F,&01,& [6924]
20,&76,&65,&72,&73,&74,&65,&63,&6B,&74,&65
,&73,&20,&47,&65,&6C,&64,&0A,&0D,&09,&09,&
FD, 2169
880 DATA &20,&41,&75,&74,&6F,&2C,&6D,&69,& [5721]
74,&20,&64,&65,&6D,&20,&73,&69,&65,&20,&66
,&61,&68,&72,&65,&6E,&0A,&0D,&09,&09,&0F,&
03, 2229
890 DATA &FC,&0F,&01,&20,&47,&61,&6E,&67,& [7124]
73,&74,&65,&72,&2C,&64,&65,&6E,&20,&73,&69
,&65,&20,&61,&6E,&68,&65,&75,&65,&72,&6E,&
20, 2753
900 DATA &6B,&6F,&65,&6E,&6E,&65,&6E,&0A,& [5277]
0D,&09,&09,&90,&20,&76,&6F,&6E,&20,&64,&65
,&72,&20,&50,&6F,&6C,&69,&7A,&65,&69,&20,&
75, 2566
910 DATA &6D,&73,&74,&65,&6C,&6C,&74,&65,& [4729]
73,&20,&47,&65,&62,&69,&65,&74,&0A,&0A,&0D
,&09,&2D,&64,&61,&73,&20,&41,&75,&74,&6F,&
20, 2485
920 DATA &77,&69,&72,&64,&20,&6D,&69,&74,& [5372]
20,&4A,&6F,&79,&73,&74,&69,&63,&6B,&20,&6F
,&64,&65,&72,&20,&20,&20,&20,&20,&20,&20,&
20, 2394
930 DATA &20,&43,&75,&72,&73,&6F,&72,&74,& [6242]
61,&73,&74,&65,&6E,&20,&67,&65,&73,&74,&65
,&75,&65,&72,&74,&0A,&0A,&0D,&09,&2D,&64,&
75, 2645
940 DATA &72,&63,&68,&20,&64,&72,&75,&65,& [5896]
63,&6B,&65,&6E,&20,&64,&65,&73,&20,&46,&65
,&75,&65,&72,&6B,&6E,&6F,&70,&66,&65,&73,&
20, 2871
950 DATA &6F,&64,&65,&72,&20,&20,&20,&20,& [7726]
64,&65,&72,&20,&43,&6F,&70,&79,&74,&61,&73
,&74,&65,&20,&6B,&6F,&65,&6E,&6E,&65,&6E,&
20, 2671
960 DATA &73,&69,&65,&20,&64,&61,&73,&20,& [6366]
53,&70,&69,&65,&6C,&20,&20,&20,&20,&20,&61
,&6E,&68,&61,&6C,&74,&65,&6E,&0A,&0A,&0D,&
09, 2251
970 DATA &2D,&62,&65,&65,&6E,&64,&65,&6E,& [6239]
20,&73,&69,&65,&20,&64,&61,&73,&20,&53,&70
,&69,&65,&6C,&2C,&6B,&6F,&65,&6E,&6E,&65,&
6E, 2798
980 DATA &20,&73,&69,&65,&20,&65,&73,&20,& [6382]
20,&20,&20,&64,&75,&72,&63,&68,&20,&43,&41
,&4C,&4C,&20,&26,&35,&38,&31,&41,&20,&77,&
69, 2128
990 DATA &65,&64,&65,&72,&20,&61,&75,&66,& [6298]
72,&75,&66,&65,&6E,&0A,&0A,&0D,&09,&09,&09
,&09,&09,&09,&09,&0F,&02,&42,&69,&74,&74,&
65, 1925
1000 DATA &20,&65,&69,&6E,&65,&20,&54,&61, [5893]
&73,&74,&65,&20,&64,&72,&75,&65,&63,&6B,&6
5,&6E,&0F,&01,&FF,&0C,&0F,&03,&0A,&0A,&09,
&09, 2214
1010 DATA &09,&09,&09,&09,&09,&09,&09,&09, [5734]
&09,&09,&09,&09,&09,&09,&47,&41,&4E,&47,&5
3,&54,&45,&52,&0F,&01,&0A,&0A,&0A,&0D,&09,
&09, 806
1020 DATA &09,&09,&53,&69,&65,&20,&68,&61, [6820]
&62,&65,&6E,&20,&69,&68,&72,&65,&20,&53,&6
1,&63,&68,&65,&20,&67,&75,&74,&20,&67,&65,
&6D, 2534
1030 DATA &61,&63,&68,&74,&21,&0A,&0A,&0D, [5593]
&09,&09,&09,&09,&09,&09,&09,&09,&09,&09,&5
3,&69,&65,&20,&64,&75,&65,&72,&66,&65,&6E,
&20, 1670
1040 DATA &73,&69,&63,&68,&20,&69,&6E,&20, [6348]
&20,&20,&0A,&0A,&08,&08,&08,&08,&08,&08,&0
8,&08,&08,&08,&08,&08,&08,&08,&64,&69,&65,
&0A, 1214
1050 DATA &0A,&0D,&09,&09,&09,&09,&09,&09, [5870]
&09,&09,&54,&6F,&70,&2D,&54,&65,&6E,&20,&6
4,&65,&72,&20,&70,&6F,&6C,&69,&7A,&65,&69,
```

```
&6C, 1993
1060 DATA &69,&63,&68,&65,&6E,&0A,&0A,&0A, [6345]
&0D,&09,&09,&09,&09,&09,&09,&09,&09,&46,&6
1,&68,&6E,&64,&75,&6E,&67,&73,&6C,&69,&73,
&74, 2004
1070 DATA &65,&20,&65,&69,&6E,&74,&72,&61, [6886]
&67,&65,&6E,&21,&0A,&0A,&0A,&0D,&09,&09,&0
9,&09,&42,&69,&74,&74,&65,&20,&67,&65,&62,
&65, 2141
1080 DATA &6E,&20,&73,&69,&65,&20,&69,&68, [4951]
&72,&65,&6E,&20,&4E,&61,&6D,&65,&6E,&20,&6
5,&69,&6E,&3A,&0F,&03, 0A,&0A,&0A,&0A,&0A,
&0D, 2043
1090 DATA &09,&09,&09,&09,&09,&09,&09,&09, [6295]
&09,&2D,&2D,&2D,&2D,&2D,&2D,&2D,&2D,&2D,&2
D,&2D,&2D,&2D,&2D,&2D,&2D,&2D,&2D,&2D,&2D,
&0F, 996
1100 DATA &01,&FF,&0C,&0A,&0A,&0F,&03,&09, [6673]
&09,&09,&09,&09,&09,&09,&09,&09,&09,&09,&0
9,&09,&09,&09,&09,&47,&41,&4E,&47,&53,&54,
&45, 971
1110 DATA &52,&0F,&01,&0A,&0A,&0A,&0A,&0D, [4837]
&53,&69,&65,&20,&68,&61,&62,&65,&6E,&20,&6
5,&73,&20,&6E,&69,&63,&68,&74,&20,&67,&65,
&73, 2147
1120 DATA &63,&68,&61,&66,&66,&74,&20,&73, [5580]
&69,&63,&68,&20,&65,&69,&6E,&65,&6E,&0A,&0
A,&0D,&50,&6C,&61,&74,&7A,&20,&69,&6E,&20,
&64, 2569
1130 DATA &65,&72,&20,&46,&61,&68,&6E,&64, [6208]
&75,&6E,&67,&73,&6C,&69,&73,&74,&65,&20,&6
4,&65,&72,&20,&50,&6F,&6C,&69,&7A,&65,&69,
&0A, 2839
1140 DATA &0A,&0D,&09,&09,&09,&09,&09,&09, [7858]
&09,&09,&09,&09,&09,&09,&09,&09,&7A,&75,&2
0,&73,&69,&63,&68,&65,&72,&6E,&21,&0A,&0A,
&0D, 1234
1150 DATA &09,&09,&53,&63,&68,&61,&6C,&74, [6159]
&65,&6E,&20,&73,&69,&65,&20,&6A,&65,&74.&7
A,&74,&20,&61,&62,&65,&72,&20,&6E,&69,&63,
&68, 2674
1160 DATA &74,&20,&67,&6C,&65,&69,&63,&68, [6372]
&0A,&0A,&0D,&09,&09,&09,&09,&09,&09,&09,&0
9,&09,&09,&09,&64,&65,&6E,&20,&43,&6F,&6D,
&70, 1642
1170 DATA &75,&74,&65,&72,&20,&61,&75,&73, [6554]
&21,&21,&0A,&0A,&0D,&09,&09,&09,&09,&09,&0
9,&09,&50,&72,&6F,&62,&69,&65,&72,&65,&6E,
&20, 1937
1180 DATA &73,&69,&65,&20,&65,&73,&20,&6E, [5421]
&6F,&63,&68,&6D,&61,&6C,&21,&21,&0F,&02,&0
A,&0A,&0A,&0A,&0A,&0A,&0D,&09,&09,&09,&09,
&09, 1540
1190 DATA &09,&09,&42,&69,&74,&74,&65,&20, [6279]
&65,&69,&6E,&65,&20,&54,&61,&73,&74,&65,&2
0,&64,&72,&75,&65,&63,&6B,&65,&6E,&0F,&01,
&FF, 2668
1200 DATA &0F,&03,&0A,&0A,&09,&09,&09,&09, [5725]
&09,&09,&09,&09,&09,&09,&09,&09,&09,&09,&0
9,&09,&09,&54,&6F,&70,&2D,&54,&65,&6E,&0F,
&03, 856
1210 DATA &0A,&0A,&0A,&0A,&0A,&0A,&0A,&0A, [5731]
&0A,&0A,&0A,&0A,&0A,&0A,&0A,&0A,&0A,&0A,&0
A,&0A,&0F,&02,&0A,&0D,&09,&09,&09,&09,&09,
&09, 294
1220 DATA &09,&42,&69,&74,&74,&65,&20,&65, [5501]
&69,&6E,&65,&20,&54,&61,&73,&74,&65,&20,&6
4,&72,&75,&65,&63,&6B,&65,&6E,&0F,&01,&FF,
&01, 2660
1230 DATA &00,&00,&54,&03,&0F,&07,&03,&00, [4587]
&02,&00,&00,&06,&00,&01,&07,&03,&00,&03,&0
0,&00,&32,&00,&00,&07,&04,&00,&01,&00,&00,
&90, 340
1240 DATA &00,&00,&05,&0F,&00,&01,&00,&00, [5183]
&64,&00,&00,&05,&08,&00,&01,&00,&00,&DE,&0
1,&00,&07,&19,&00,&01,&00,&00,&66,&01,&00,
&07, 501
1250 DATA &14,&00,&01,&00,&00,&00,&00,&00, [5543]
&07,&05,&00,&01,&00,&00,&66,&01,&00,&07,&0
C,&00,&01,&00,&00,&3F,&01,&00,&07,&0C,&00,
&01, 241
1260 DATA &00,&00,&1C,&01,&00,&07,&19,&00, [5679]
&01,&00,&00,&66,&01,&00,&07,&19,&00,&01,&0
0,&00,&EF,&00,&00,&07,&32,&00,&01,&00,&00,
&1C, 523
1270 DATA &01,&00,&07,&32,&00,&FF,&0C,&0A, [5481]
&0A,&0F,&03,&09,&09,&09,&09,&09,&09,&09,&0
9,&09,&09,&09,&09,&09,&09,&09,&09,&47,&41,
&4E, 721
1280 DATA &47,&53,&54,&45,&52,&0F,&01,&0A, [5931]
&0A,&0A,&0A,&0A,&0A,&0D,&09,&09,&09,&09,&0
9,&09,&09,&09,&09,&09,&47,&65,&62,&65,&6E,
&20, 1081
1290 DATA &73,&69,&65,&20,&62,&69,&74,&74, [4566]
&65,&20,&64,&69,&65,&0A,&0A,&0D,&09,&09,&0
9,&09,&09,&09,&09,&09,&53,&74,&61,&72,&74,
&67, 1961
1300 DATA &65,&73,&63,&68,&77,&69,&6E,&64, [6522]
&69,&67,&6B,&65,&69,&74,&20,&61,&6E,&0F,&0
2,&0A,&0A,&0A,&0A,&0A,&0D,&09,&09,&09,&09,
&09, 1854
1310 DATA &09,&09,&09,&09,&09,&09,&09,&09, [5798]
&3C,&31,&3E,&0F,&01,&20,&20,&20,&31,&30,&2
0,&20,&6B,&6D,&2F,&68,&0F,&02,&0A,&0A,&0D,
&09, 942
1320 DATA &09,&09,&09,&09,&09,&09,&09,&09, [4429]
&09,&09,&09,&09,&3C,&32,&3E,&0F,&01,&20,&2
0,&20,&35,&30,&20,&20,&6B,&6D,&2F,&68,&0F,
&02, 941
1330 DATA &0A,&0A,&0D,&09,&09,&09,&09,&09, [7123]
&09,&09,&09,&09,&09,&09,&09,&09,&3C,&33,&3
E,&0F,&01,&20,&20,&20,&39,&30,&20,&20,&6B,
&6D, 820
1340 DATA &2F,&68,&0F,&02,&0A,&0A,&0D,&09, [5676]
&09,&09,&09,&09,&09,&09,&09,&09,&09,&09,&0
9,&09,&3C,&34,&3E,&0F,&01,&20,&20,&20,&31,
&33, 704
1350 DATA &30,&20,&6B,&6D,&2F,&68,&FF,&0D, [5593]
&09,&09,&09,&09,&09,&31,&30,&0B,&0D,&09,&0
9,&09,&09,&09,&39,&0B,&0D,&09,&09,&09,&09,
&09, 1052
1360 DATA &38,&0B,&0D,&09,&09,&09,&09,&09, [5409]
&37,&0B,&0D,&09,&09,&09,&09,&09,&36,&0B,&0
D,&09,&09,&09,&09,&09,&35,&0B,&0D,&09,&09,
&09, 476
1370 DATA &09,&09,&34,&0B,&0D,&09,&09,&09, [6638]
&09,&09,&33,&0B,&0D,&09,&09,&09,&09,&09,&3
2,&0B,&0D,&09,&09,&09,&09,&09,&31,&FF, 682
```









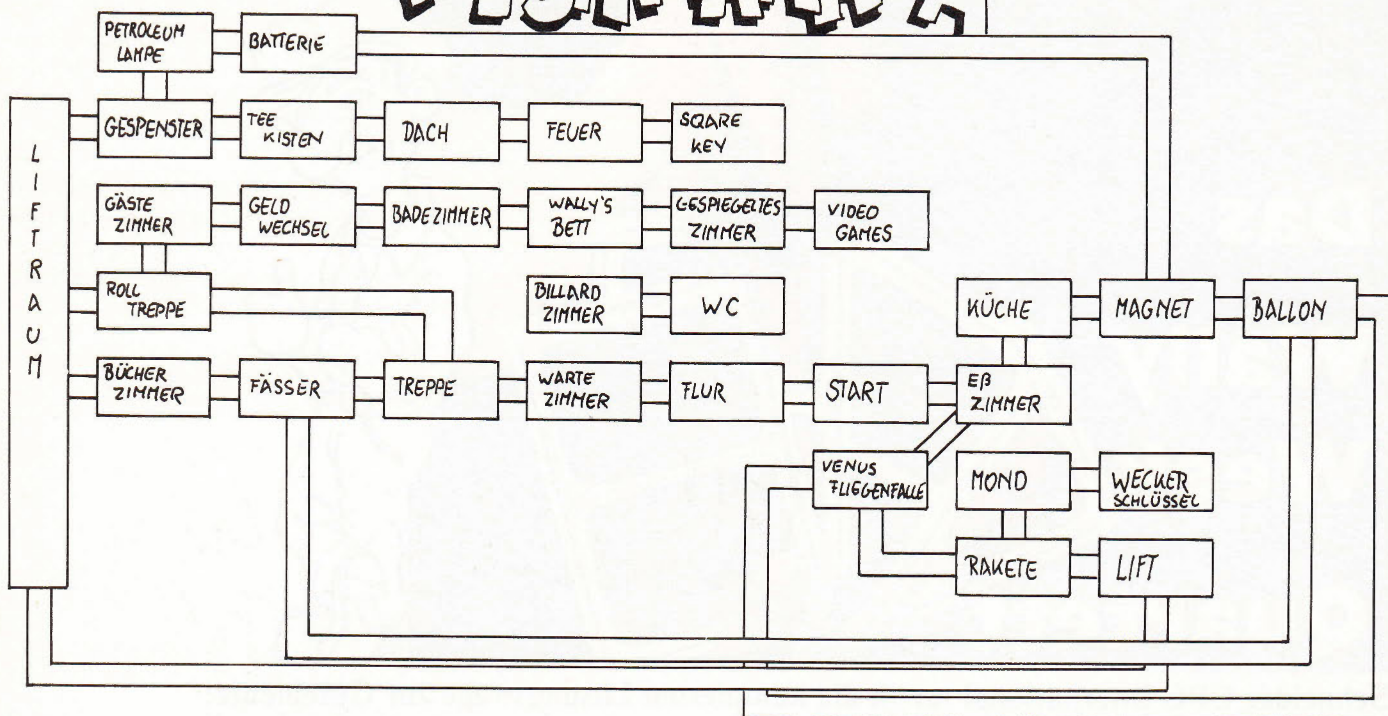
Schneider CPC International verrät die kompletten Lösungswege zur Geschichte:

Eine der ersten Firmen, die Programme für Schneider/Amstrad Computer umsetzte, war die englische Softwarefirma Mikro Gen.
Das erste Programm für die CPC's war Pyjamarama. Es lief zuvor schon sehr erfolgreich auf anderen Computern an und zeichnet sich durch gute Grafik- und Toneffekte aus.

Bemerkenswert ist die große Anzahl an Räumen, die man erkunden kann. Dieses Grundkonzept hat Mikro Gen auch bei den Nachfolgeprogrammen (Automania, Everyone's a Wally, Herbert's Dummy Run und Three Weeks in Paradise) fortgeführt. Doch leider gelingt es nicht vielen Spielern, jeweils das Ziel zu erreichen, da man für das Ausknobeln des Lösungsweges sehr viel Zeit braucht. Jeder Gegenstand, der im Programm auftaucht, hat nämlich eine bestimmte Funktion. Hat man diese herausgefunden, muß sie in Einklang mit dem gesamten Programm gebracht werden....
Zusätzlich zu den Lösungswegen veröffentlichen wir hier noch Lagepläne zu den Programmen, auf denen alle wichtigen Räume eingezeichnet und benannt sind, damit Sie es noch einfacher haben, die Programme nun zu bewältigen.
Der Titel dieses Berichts ist ein bißchen irreführend, weil es sich insgesamt um fünf Programme handelt. Der hier verwendete Name "Quartett" ist insofern gerechtfertigt, weil "Automania" nicht dem Arcade-Action-Typ zugehört. Hier benötigt man keinen speziellen Lösungsweg. Dabei hat Wally die Aufgabe, 10 Autos zu bauen, wovon jedes aus 6 Teilen besteht. Es geht im Verkaufsraum los, in dem auf die umhersausenden Reifen aufgepaßt werden muß, welche Wally treffen wollen. Anschließend muß Wally im Lager die richtigen Teile suchen und die Leiter hochsteigen, um sie zu holen. Überall liegt etwas herum, über das Wally fallen könnte und deshalb darüberspringen muß.
Dann baut man mit Wally in der Werkstatt die Teile in die richtige Position, wobei für das Dach und die Windschutzscheibe spezielle "Hochsprünge" nötig sind. Soviel zum Spielprogramm "Automania", das – wie alle bisher von Mikro Gen erschienenen Programme – für jede Spielernatur sehr empfehlenswert ist.

Pyjamarama

Für diejenigen, die "Pyjamarama" noch nicht besitzen, soll hier einmal ganz kurz der Spielsinn dieses Programms, das eher ein Grafik-Adventure als ein (Action-) Game ist, genannt werden (siehe auch Bild 1):
Das Spielziel:
Die Hauptfigur des Programms ist Wally Week, ein notorischer Langschläfer. Die Aufgabe des Spielers ist es, ihn mit Hilfe eines Weckers aufzuwecken, damit Wally pünktlich zur Arbeit kommt.
Doch bevor der Wecker überhaupt funktioniert, muß er mit einem Schlüssel aufgezogen werden. Dieser Aufziehschlüssel befindet sich auf dem Mond. Aber leider hat das Bild, welches nach dem Mond kommt, eine magnetische Sicherung. Diese kann nur mit einem Magneten aufgehoben werden. Soviel zuerst zu "Pyjamarama". Zur weiteren Hilfe sollen hier noch ein paar Pokes genannt werden, mit denen man mehr von seinem bzw. Wallys Leben hat:
Die Unsterblichkeit:
Mit POKE 16087,0 vor der letzten Ladeanweisung im Basic-Lader (LOAD "!Name") ist man unsterblich. Verluste der "Schnarchenergie" können aber auch durch POKE &23E4,0 :POKE &23F8,0 verhindert werden.
Sind einige dieser POKEs an verschiedenen Stellen in Ihrem Programm eingebaut, dann können Sie sich darauf verlassen, daß es Ihnen nun gelingt, "Pyjamarama" zu lösen.
Dazu gehen Sie wie folgt vor (siehe dazu auch Lageplan 1):
Zur Vorgehensweise:
Bevor man sich auf den untenstehenden Lösungsweg begibt, sollte man ein wenig durch alle Räume wandern, damit man ungefähr einen Plan des Hauses hat. Dies empfiehlt sich auch bei den übrigen Lösungswegen bzw. Programmen.



Jetzt geht's los:

1. Besorgen Sie sich den BUCKET (Wassereimer) und füllen Sie Ihn im Badezimmer am Wasserhahn mit Wasser.

2. Besorgen Sie sich jetzt den PLANT POT (Blumentopf). Gehen Sie dann mit dem Blumentopf und dem gefüllten Eimer in den Raum, in dem sich die "Venus-Fliegenfallen" befinden. Jetzt sind nämlich diese Fliegenfallen außer Gefecht gesetzt und Sie können sehr leicht den BP CAN (BP Kanister) einsammeln, müssen aber dafür den Eimer zurücklassen.

3. Hierauf muß der Kanister gefüllt werden. Gehen Sie anschließend in den Liftraum und berühren Sie die dort befindliche Zahl "3". So gelangen Sie in den Raketenraum. In der Nähe der Rakete liegt der TRIANGLE KEY (Dreiecks-Schlüssel). Nehmen Sie diesen auf!

Gehen Sie dann mit dem Kanister und dem TRIANGLE KEY in den Liftraum zurück. Wenn Sie dort angekommen sind, springen Sie an die Zahl "1".

Hiernach gehen Sie durch die Lifttür nach rechts durch die Tür, die als nächste kommt. In dem Raum, in dem Sie sich jetzt befinden, können Sie an der Petroleumpumpe den BP Kanister auffüllen.

4. Auf dem Mond befinden sich Aliens. Möchte man ohne Schaden an diesen Kreaturen vorbeikommen, dann muß man ein geladenes Lasergewehr besitzen. Dies erhält man folgendermaßen:

4.a. Wechseln Sie das POUND gegen den PENNY ein.

4.b. Tauschen Sie den Penny gegen den Hammer. Gehen Sie nun in den Liftraum und tauschen Sie dort den Hammer gegen den FIRE EXTINGUISHER (Feuerlöscher) ein.

4.c. Verlassen Sie den Liftraum durch Tür 1. Gehen Sie jetzt in den Raum, in dem Teekisten aufeinander gestapelt sind. Dort geistern auch "Gespenster" herum. Von hier können Sie durch das obere Fenster entweichen. Wenn Sie nun rechts entlang gehen, kommen Sie in einen Raum, in dem sich ein "kaltes Feuer" befindet. Mit dem Feuerlöscher können Sie dieses Feuer problemlos durchqueren. Nun müssen Sie weiter nach rechts gehen und nachdem Sie gefallen sind, sollten Sie schnell nach links gehen und den SQUARE KEY einkassieren. An dieser Stelle müssen Sie sich beeilen, da Sie sonst von einem riesigen Ball plattgewalzt werden.

4.e. Mit dem SQUARE KEY kommen Sie in den Billardraum. Hier finden Sie endlich das Lasergewehr (LASER GUN).

4.f. Jetzt müssen Sie in den Liftraum zurückwandern und dort durch Tür 1 gehen. Vergewissern Sie sich aber vorher, ob Sie den TRIANGLE KEY und das LASER GUN bei sich haben. Nun geht es rechts lang durch die nächste Tür; Sie befinden sich jetzt in dem Raum mit der Petroleumpumpe. Von hier aus geht es weiter durch die rechte Tür. In diesem Raum liegt auf einem Tisch eine Batterie. Nehmen Sie diese Batterie kurz auf und stellen Sie diese danach sofort wieder ab. Jetzt müssen Sie sich durch den unteren Ausgang fallen lassen.

Endlich, endlich, endlich! Sie sind nun stolzer Besitzer eines geladenen Lasergewehrs (LASER GUN).

Doch das Programmziel ist noch immer nicht erreicht!

4.g. Nehmen Sie das Lasergewehr (LASER GUN) und besorgen Sie sich den gefüllten BP Kanister, den Sie irgendwo zurückgelassen haben. Mit beiden Teilen, LASER GUN und BP CAN, müssen Sie nun zum Raketenkeller gehen und dort in die Rakete einsteigen. Durch diese Rakete werden Sie dann zum Mond befördert. Lassen Sie auf dem Mond das Lasergewehr zurück und fliegen Sie wieder zur Erde. Dort wieder angekommen, gilt es, den Magneten zu besorgen, um damit die magnetische Sperre zu knacken.

5. Es geht weiter, indem Sie sich die DRIVING LICENCE besorgen; diese liegt im ersten Gespensterraum. Den Führerschein (DRIVING LICENCE) können Sie dann gegen die IGNITION KEYS, die sich auf dem Dach befinden, eintauschen. Im Treppenhaus rutschen Sie später das Treppengeländer herunter und kommen so zum CRASH HELMET

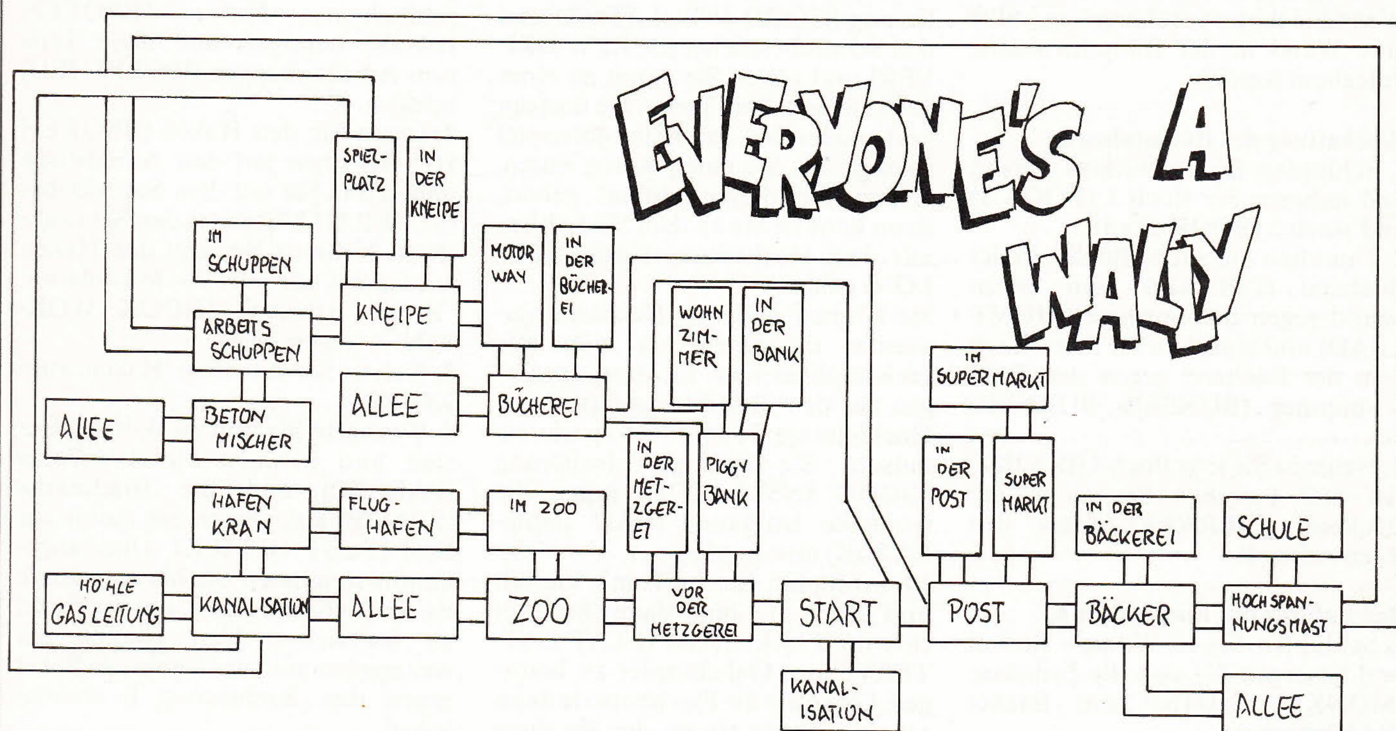
(Sturzhelm). Im Besitz dieses Sturzhelms holen Sie sich das LIBRARY TICKET und tauschen es gegen das LIBRARY BOOK ein. Durch die Kombination von CRASH HELMET und LIBRARY BOOK kommen Sie in die Lage, die SCISSORS (Schere) aufzunehmen.

6. Stellen Sie den HELP BUTTON in dem Raum, in dem der Geldwechsler steht, auf ON und machen Sie sich auf den Weg zu dem Raum, in dem ein Luftballon hängt. Oben unter der Decke befindet sich auch der BOX KEY (Box-Schlüssel). Um diesen Schlüssel zu bekommen, müssen Sie den Ballon berühren und schweben dann mit diesem nach oben. Dadurch, daß die HELPFUNKTION eingeschaltet ist, müßte mittlerweile unter dem Box-Schlüssel eine zusätzliche Teekiste stehen. Sie ermöglicht es Ihnen, den BOX KEY zu erreichen.
Falls dort keine zusätzliche Teekiste steht, müssen Sie die HELP-FUNKTION erneut einschalten.

7. Gehen Sie jetzt in die Küche und schließen Sie mit dem BOX KEY das kleine Kistchen auf. In diesem Kistchen befindet sich der Magnet. Nehmen Sie diesen Magneten mit!

8. Ab jetzt ist das restliche Geschehen sehr einfach: Fliegen Sie mit dem Magneten zum Mond zurück und holen Sie sich den Weckerschlüssel. Berühren Sie dann abschließend mit diesem in Wallys Zimmer den Wecker.

Das Spielende naht
Dieser klingelt dann so laut, daß Wally wach wird und es endlich einmal schafft, dank Ihrer unermüdlichen Mithilfe, pünktlich zur Arbeit zu kommen!!!



Everyone's a Wally

Das zweite Programm der Wally-Serie ist etwas schwieriger zu lösen, da die anderen Figuren bzw. Charakter Ihnen fortwährend begegnen und Sie auch in deren Rolle schlüpfen müssen. Welche Person Sie derzeitig sind, und wo sich die übrigen Figuren momentan aufhalten, kann man aber glücklicherweise durch Tastendruck (siehe Lageplan 2) erfahren. Sollten Sie irgendwelche Probleme haben, dann müssen Sie nur weiterlesen, um sie zu lösen.

Zum Spiel:
Der Untertitel dieses Programms wäre "Ein Tag im Leben von Wally".
"Everyone's a Wally" kann man als das erste Multirollen-Arkadeabenteuerspiel bezeichnen. Neben dem bekannten Wally Week treten hier persönlich folgende Figuren zum ersten Mal auf: Wilma, Wally's Frau, Herbert, Wally's Sohn und die Assistenten Tom, Dick und Harry (siehe auch Bild 2). Während des ganzen Spiels kann man die Kontrolle auf eine andere der fünf Personen übernehmen. Jede Person hat eine eigene Aufgabe zu spielen und hierbei muß der Spieler entscheiden, welche Person in der Lage ist, eine bestimmte Aufgabe zu erledigen. Diese Überlegung gibt Ihnen aber unser Lösungsweg preis.
Herbert ist hier, wie alle kleinen Kinder, überhaupt nicht kontrollierbar. Zur musikalischen Untermalung empfiehlt es sich, eventuell die auf der Kassettenrückseite mitgelieferte Hit-Single "Everyone's a Wally" von Mike Berry anzuhören. Das Lied erschien sogar auch auf einer Schallplatte!

Der Start:
Um den Springbrunnen zu reparieren, müssen Sie sich die Erdnüsse (MONKEY NUTS) vom Bäcker (BAKER) besorgen, indem Sie Wilma verwenden. Nun können Sie in den Zoo eintreten, die Klempnerzange (MONKEY WRENCH) aufnehmen und an einer Stelle, an der Sie sie später wieder leicht finden und aufnehmen können, ablegen.
Beachten Sie bitte, daß auch eine der übrigen Personen einen bestimmten Gegenstand, den Sie brauchen und daher suchen, besitzen kann.
Dies können Sie herausfinden, indem Sie einfach einmal kurz in jede Gestalt hineinschlüpfen!
Suchen Sie nun Dick, den Klempner, und nehmen Sie mit ihm seine Saugglocke (PLUNGER) auf. Nehmen Sie nun die zuvor abgelegte Klempnerzange wieder auf und gehen Sie mit Dick zum Springbrunnen (versichern Sie sich, daß Sie beides, die Klempnerzange und die Saugglocke, bei sich haben).
Springen Sie auf den Springbrunnen und dann auf die Spitze der Statue, wodurch der Brunnen repariert wird.

Ist der Springbrunnen wieder funktionstüchtig, dann sollten Sie durch Wally den leeren Eimer (EMPTY BUCKET) und den Sand (sand) aufnehmen. Gehen Sie mit Wally anschließend zum Springbrunnen. Der Eimer wird dann gefüllt (FULL EMPTY).
Haben Sie dies gemacht, dann müssen Sie mit Wally zur Zementmischmaschine (CEMENT MIXER) gehen und den Zement (CEMENT) aufnehmen. Darauf ist es Ihre Aufgabe, die Maurerkelle (TROWEL) zu suchen. Gehen Sie dann mit der Maurerkelle und dem Zement zu dem Ort, an dem die Ziegelsteine (BRICKETS) liegen. Wenn Sie daran vorübergehen, sollte eine Wand in der Bildschirmmitte aufgebaut werden.

Beschaffung des Buchstaben B:
1. Schlüpfen Sie in Wilmas Person und nehmen Sie Buch 1 (BOOK 1) und Buch 2 (BOOK 2) auf.
2. Tauschen Sie zuerst Buch 1 in der Bücherei (LIBRARY) am ersten Schild gegen das Springseil (JUMP LEAD) und tauschen Sie dann Buch 2 in der Bücherei gegen den Bunsenbrenner (BUNSEN BURNER) ein.
3. Nehmen Sie jetzt Buch 3 (BOOK 3) auf und tauschen Sie es in der Bücherei (LIBRARY) gegen den Buchstaben B.

Beschaffung des Buchstaben K:
1. Schlüpfen Sie in Wilma's Person und besorgen Sie sich die Erdnüsse (MONKEY NUTS) vom Bäcker (BAKER).
2. Gehen Sie mit den Erdnüssen in den Zoo und nehmen Sie dort die Klempnerzange auf.
Legen Sie diese dann an einer Stelle ab, die Sie später leicht wiederfinden.
3. Wechseln Sie nun in Dick's Gestalt und gehen Sie durch die Räume, um sich die Saugglocke (PLUNGER) zu besorgen. Nehmen Sie nun auch die Klempnerzange (Monkey Wrench) wieder auf.
4. Reparieren Sie mit Dick den Springbrunnen, indem Sie auf die Spitze des Springbrunnens steigen bzw. springen. Dazu müssen Sie aber beides, die Klempnerzange (MONKEY WRENCH) und die Saugglocke (PLUNGER) besitzen.
5. Schlüpfen Sie wieder in Wally's Person und nehmen Sie so den leeren Eimer (EMPTY BUCKET) und den Sand (SAND) auf.
6. Füllen Sie den Eimer am Springbrunnen und gehen Sie dann weiter, um den Zement (CEMENT) in dem Bildschirm, in dem sich die Zementmischmaschine (CEMENT MIXER) befindet, aufnehmen.

7. Gehen Sie nun mit der Maurerkelle (TROWEL) und dem Zement (CEMENT) zum Raum mit den Ziegelsteinen. Laufen Sie nun an diesen vorbei, wodurch eine Wand errichtet wird.
8. Wechseln Sie jetzt in Harry's Rolle und nehmen Sie mit ihm den Sicherungsdraht (FUSE WIRE) und die durchgebrannte Sicherung (BLOWN FUSE) auf, wodurch diese wieder repariert wird.
9. Besorgen Sie sich jetzt die gute Isolierung (GOOD INSULATOR) und den Schraubenzieher (SCREW DRIVER) und gehen Sie damit zu einer der Telefonzellen. Treten Sie dort ein und spielen Sie das Asteroidenspiel solange, bis Sie einen Klang hören. Haben Sie diesen einmal gehört, dann können Sie zu dem Bildschirm mit dem Hochleitungsmasten (PYLON) gehen.
Sie können nun den Hochleitungsmasten reparieren, da sich alle Lichtstrahlen bzw. Elektrospannungen aus dem Bild bewegen (um den Hochleitungsmasten zu reparieren, müssen Sie die gute Isolierung (GOOD INSULATOR) gegen die schlechte Isolierung (BAD INSULATOR) austauschen.
10. Schlüpfen Sie in Tom's Gestalt und gehen Sie dann damit fort, um sich die Flachbatterie (FLAT BATTERY) vom Gabelstapler zu besorgen. Legen Sie die Flachbatterie dann an einer Stelle ab, an der Sie diese später leicht wiederfinden können.

11. Wechseln Sie nun in Harry's Rolle und holen Sie die Flachbatterie (FLAT BATTERY) wieder zurück. Gehen Sie dann mit dieser und dem Springseil (JUMP LEAD) zur Bee Pee-Station.
12. Wenn Sie die Flachbatterie hier einmal mit der grünen Batterie, die sich in der Bee Pee-Station befindet, aufgeladen, haben dann müssen Sie die jetzt wieder aufgefüllte Batterie an einer Stelle ablegen, an der Sie diese später leicht wiederfinden und schnell aufnehmen können.
13. Schlüpfen Sie wieder in Tom's Charakter und nehmen Sie die Flachbatterie (FLAT BATTERY) auf.
Bringen Sie diese darauf zum Gabelstapler zurück. Wechseln Sie dann in Wally's Rolle und steigen Sie in den Gabelstapler ein.
Klettern Sie danach auf die Wand und nehmen Sie den Buchstaben K auf.

Beschaffung des Buchstaben A:
1. Schlüpfen Sie in Tom's Gestalt und besorgen Sie sich die Ölkanne (OIL CAN). Gehen Sie damit zum Supermarkt. Ölen Sie dort den Einkaufswagen.
2. Begeben Sie sich jetzt in Harry's Rolle, springen Sie auf den Einkaufswagen und dann auf die Ladentheke und nehmen Sie dort den Buchstaben A auf.

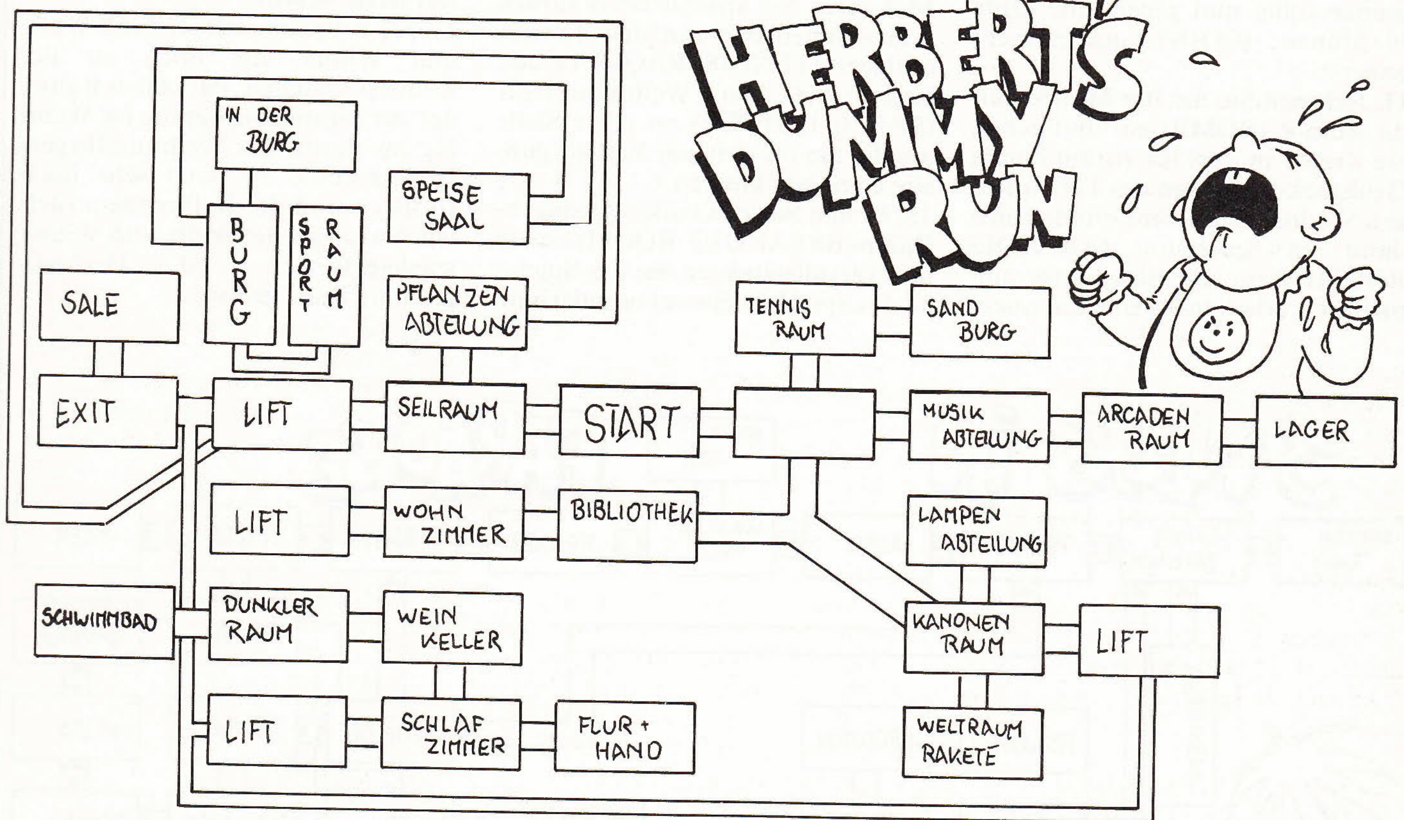
Beschaffung des Buchstaben E:
1. Um den Haken (HOOK) zu reparieren, müssen Sie sich in Wally's Gestalt befinden, sich den Superkleber (SUPER GLUE) besorgen und den gebrochenen Haken (BROKEN HOOK) besorgen und diese Teile zum Arbeitsschuppen (WORKSHED) bringen.
2. Legen Sie den Haken (BROKEN HOOK) oben auf den Arbeitstisch und gehen Sie mit dem Superkleber (SUPER GLUE) unter den Schraubstock. Nehmen Sie jetzt den Haken wieder auf, der jetzt die Information "Haken arbeitet" (HOOK WORKING) tragen sollte.
3. Gehen Sie mit dem Haken zum Krahn.
4. Wechseln Sie nun zu Wilma's Gestalt und nehmen Sie das Paket (PARCEL) und die Briefmarke (STAMP) auf. Gehen Sie damit zur Post (POST OFFICE). Dort angekommen, müssen Sie bis zum Ende des Bildschirms gehen, um das Paket zu frankieren. Nun können Sie weitergehen und das frankierte Paket gegen den Buchstaben E austauschen.

Beschaffung des Buchstaben R:
1. Nehmen Sie die Gasmaske (GAS MASK) auf und gehen Sie, hinter dem Hai, in die Kanalisation und dann in die Höhle.
2. Gehen Sie bis zum Ende dieses Raums bzw. Bildschirms und nehmen Sie dort den Buchstaben R auf.

Reparatur der Gasleitung und Vervollständigung des Spiels:
Schlüpfen Sie in Dick's Rolle und nehmen Sie die Gasmaske (GAS MASK) auf. Gehen Sie dann zur Höhle und sammeln Sie die undichte Gasleitung (LEAKING GAS PIPE) ein (achten Sie dabei auf den gefährlichen Hai).
Wenn Sie aus der Höhle hinausgehen, müssen Sie das Kaugummi (CHEWING GUM) suchen und aufnehmen, die Gasleitung im Arbeitsschuppen (WORKSHED) auf den Arbeitstisch legen und wieder fortgehen, um das Pflaster (PATCH)

zu suchen. Nehmen Sie das Pflaster (PATCH) und das Kaugummi (CHEWING GUM) auf und gehen Sie zusammen mit der undichten Gasleitung unter den Arbeitstisch. Nehmen Sie dort die reparierte Gasleitung auf. Nehmen Sie die Gasmaske (GAS MASK) wieder auf und bringen Sie die reparierte Gasleitung zurück in die Höhle. So verschwinden die Funken in der Höhle und die Flamme von der Straße (MOTORWAY).

Das Ende naht:
Nun besitzen Sie alle Buchstaben des Codes und müssen sie zur Bank bringen. Ordnen Sie diese dort so, als wenn Sie das Word BREAK buchstabieren wollen. Jede Person sollte im Spiel einen eigenen Buchstaben einnehmen:
B muß Wilma aufnehmen,
R muß Tom aufnehmen,
E muß Dick aufnehmen,
A muß Harry aufnehmen und
K muß Wally aufnehmen,
um die Schlußgrafik zu erreichen!



Herbert's Dummy Run

Dies ist das erste Programm der Wally-Serie, in dem der kleine Herbert der Star ist. Das muß aber nicht bedeuten, daß hier nicht genausoviel Spaß entsteht wie bei den anderen Programmen, weil auch "Herbert's Dummy Run" über eine sehr gute Grafik, zahlreiche Räume und nette Effekte verfügt, die den Joystick fordern, ohne einfach "drauflos" zu ballern!

Ziel des Spiels:
Wally's Sohn Herbert hat seine Eltern verloren. Verzweifelt irrt er daher durch die Gegend – d.h. durch ein Wohnhaus, eine Burg oder sogar durch ein Kaufhaus. Dabei begegnen ihm zwar viele Gegenstände, aber leider nicht seine Eltern. Wilma und Wally warten nämlich an einer Rolltreppe auf ihren Sohn, da sie hier sein Fehlen bemerkten. Nur durch Ihre Mithilfe kann Herbert sie wiederfinden.

Der Lösungsweg:
Beachten Sie hierzu auch Lageplan 3, der die Lage und Verbindungen der wesentlichsten Räume beinhaltet.
1. Zuerst müssen Sie sich den Box-Schlüssel (BOX KEY) und den Honigtopf (HONEY POT) besorgen.
2. Hierzu gilt es, in den Raum zu gehen, in dem das Spiel begann. Springen Sie dort auf den Kasten. So gelangen Sie dann zu einer Stelle, an der Sie den Honigtopf (HONEY POT) gegen den Teddybär (TEDDY) eintauschen können.
3. Nehmen Sie das Seil (ROPE) auf und gehen Sie danach mit dem Seil und dem Teddybär zum Arcade-Raum, in dem kleine Roboter umherschwirren.
4. Der Teddybär geht nun fort und öffnet dann am rechten Bildschirmrand eine Tür, durch die Sie nun auch durchgehen können. Springen Sie jetzt an das Seil, welches sich im nächsten Raum befindet. Es verlängert sich so, daß Sie jetzt vom Schwimmbad aus mit dem Seil zur Gummiente (RUBBER DUCK) kommen und diese anschließend aufnehmen können.
5. Nehmen Sie jetzt das Flaschengewehr (POP GUN) und den Korken (CORK) auf und gehen Sie damit zur Burg.
6. Tauschen Sie das Flaschengewehr (POP GUN) gegen die Fahne (FLAG), die sich oben auf dem Burgturm befindet, ein. Gehen Sie anschließend mit der Gummiente (RUBBER DUCK) zum Start-Bildschirm. Von hier aus können Sie nun die Kieselsteine (PEBBLES) aufnehmen, um die Schleuder (CATAPULT) zu laden.
7. Besorgen Sie sich jetzt die Taschenlampe (TORCH) und die Glühbirne (BULB). Auf diese Art und Weise wird die Taschenlampe repariert und ist damit wieder funktionstüchtig.
8. Gehen Sie mit der Taschenlampe in den dunklen Raum. Hier können Sie nun alles sehen. Schießen Sie auf alle Enten; danach fällt eine Münzenrolle herunter. Nehmen Sie diese auf und legen Sie sie an einer Stelle ab, die Sie später schnell und leicht

wiederfinden können.

9. Nehmen Sie den A-Ziegelstein (A-BRICKET) und die 10p-Schokoladenmünze (CHOCOLATE COIN) auf und gehen Sie damit zu dem Bildschirm, in dem sich die Ladenkasse befindet.

10. Klettern Sie auf die Spitze des Ziegelsteins und gehen Sie mit der 10p-Schokoladenmünze an der Ladenkasse vorbei; die 10p-Schokoladenmünze sollte nun gegen eine echte 10-Münze (COIN) ausgetauscht sein.

11. Nehmen Sie die 10p-Münze und die Bombe (BOMB) auf und gehen Sie darauf in den Raum mit einem Geldstück-Schlitz an der Tür. Springen Sie dort heran. Sie befinden sich dann im Spielraum (ARCADE-ROOM). Wenn alle Blitze eingesammelt sind, erhalten Sie eine Kanonenkugel (CANNONBALL).

12. Mit der Kanonenkugel und der Münzenrolle müssen Sie nun zum Raum gehen, in dem sich die Kanone befindet. Gehen Sie durch diesen Raum. Die Kanonenkugel wird daraufhin abgeschossen und es entsteht ein Loch in der Wand.

13. Gehen Sie durch dieses Loch und nehmen Sie den Weltraumflieger (SPACE HOPPER) auf.

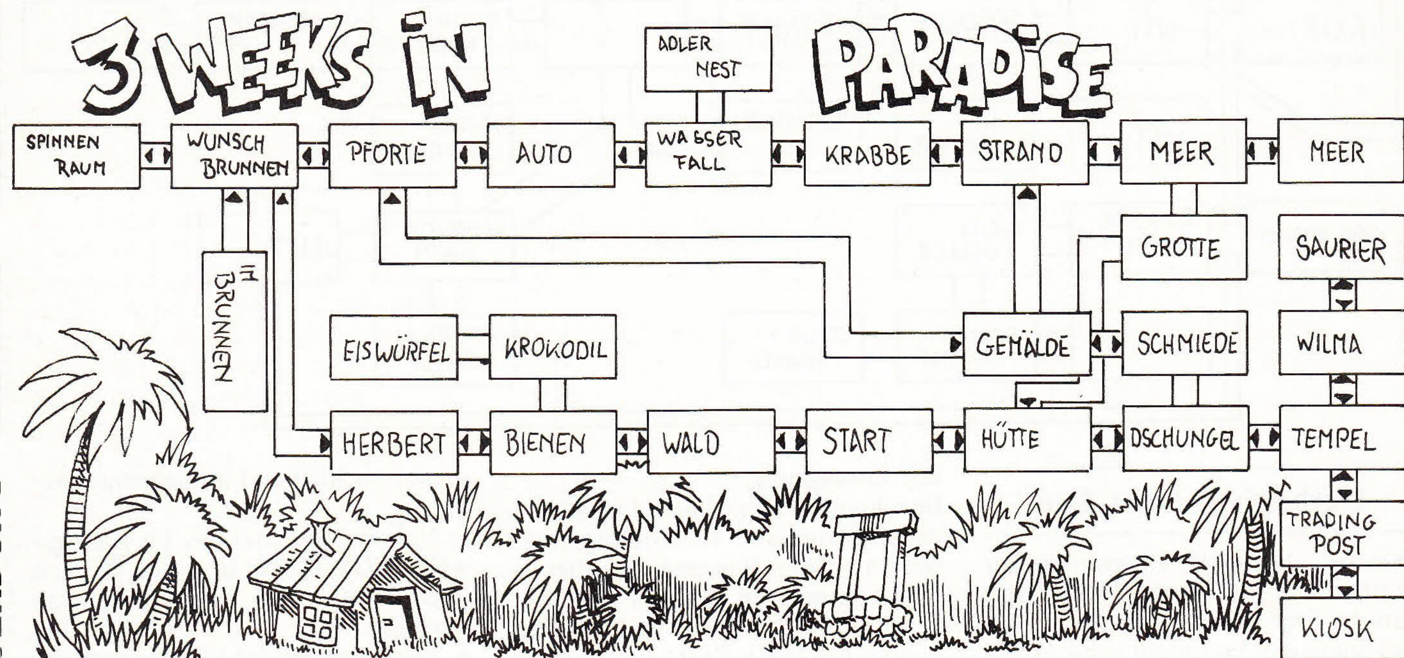
14. Laufen Sie anschließend zurück und nehmen Sie nun den Tennisschläger (TENNIS RAQUET) auf. Legen Sie den Weltraumflieger (SPACE HOPPER) an einer Stelle ab, die Sie im weiteren Verlauf günstig erreichen können.

15. Gehen Sie jetzt in den Ausbrech-Raum (BREAKOUT-ROOM) hinein und vervollständigen Sie das Spiel. Ist das Spiel hier einmal beendet bzw. vervollständigt, dann sollten Sie einen Handschuh (GLOVE) erhalten.

16. Nehmen Sie den Weltraumflieger (SPACE HOPPER) wieder auf und gehen Sie mit dem Handschuh zur Stufe 1 (Etage 1 bzw. Level 1). Hier bewacht ein Handschuh die Haustür (die Hand wird Sie nicht länger hindern, da Sie den Handschuh besitzen).

Der letzte Schritt:

17. Sie befinden sich jetzt mit Wally und Wilma, die oben an der Rolltreppe stehen, im letzten Raum, der zur Lösung notwendig ist. Wenn Sie im Besitz des Weltraumfliegers sind, können Sie jetzt sehr hoch springen und die Rolltreppe anstellen. So haben Sie Wally und Wilma wiedergefunden und "Herbert's Dummy Run" gelöst.



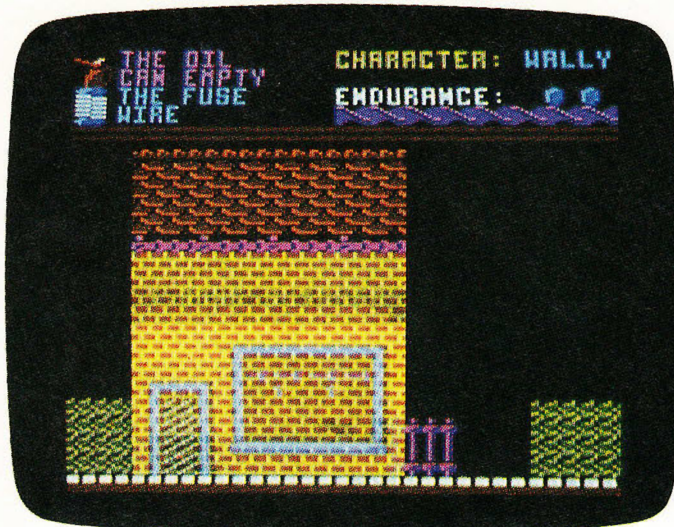
Three Weeks in Paradise

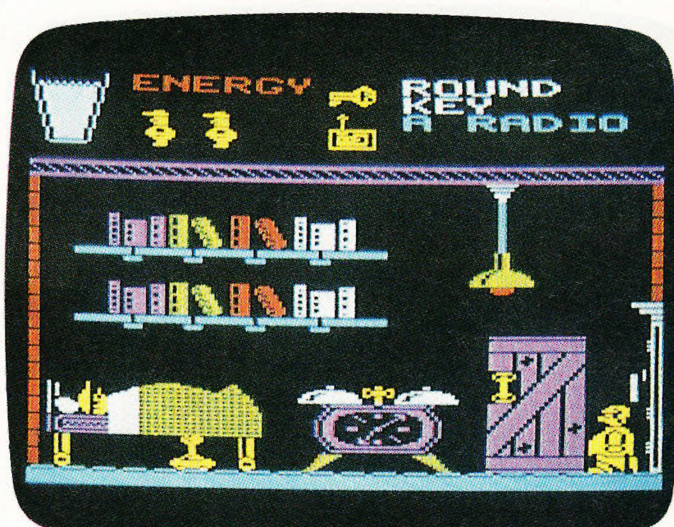
Das Programm "Three Weeks in Paradise" ist das neueste Action-Adventure der englischen Softwarefirma Mikro Gen. Schneider International beschrieb und testete dieses Programm bereits schon in Ausgabe 6/86. Hier wird nun der komplette Lösungsweg vorgestellt.

Der Spielsinn:
Wilma, Herbert und zwei Ihrer besten Freunde, sind bei ihrem diesjährigen Urlaub bei einem Streifzug durch den Dschungel von Kannibalen gefangengenommen worden.
Durch Zufall erfahren Sie davon und eilen sofort Ihren Freunden zur Hilfe. Viel Zeit bleibt Ihnen dazu nicht, da sich die Kannibalen schon auf das leckere Essen der Marke Mensch freuen!

Bevor man sich auf den untenstehenden Lösungsweg begibt, sollte man auch bei diesem Programm ein wenig durch alle Räume wandern, damit man ungefähr einen Plan des Geländes bzw. Dschungels hat und so die einzelnen Lösungsschritte schneller vollziehen kann. Die Bildschirminformationen im unteren Bildschirmteil sollten Sie genau beachten. Mit Aktion-Knopf ist gemeint, daß Sie den Joystick nach vorne drücken müssen.

Wilma's Rettung (Ladies first!):
Besorgen Sie sich zuerst das Pfefferminzbonbon (Bezeichnung am Bildschirm: MINT). Dieses finden Sie hinter dem Trading Post Schild (siehe auch Lageplan 4).
Anschließend brauchen Sie Wilmas Handtasche; dazu müssen Sie sich in den Bildschirm begeben, in dem ein großer Tisch steht. Über diesem Tisch hängt ein Gemälde, durch das Sie springen müssen. So gelangen Sie an den Strand, wo Sie Wilmas Handtasche (HANDBAG) aufnehmen müssen. Solange Sie diese Handtasche besitzen, können Sie von dem Krokodil nicht angegriffen werden.
Begeben Sie sich jetzt in den Raum, in welchem ein Eiswürfel liegt und

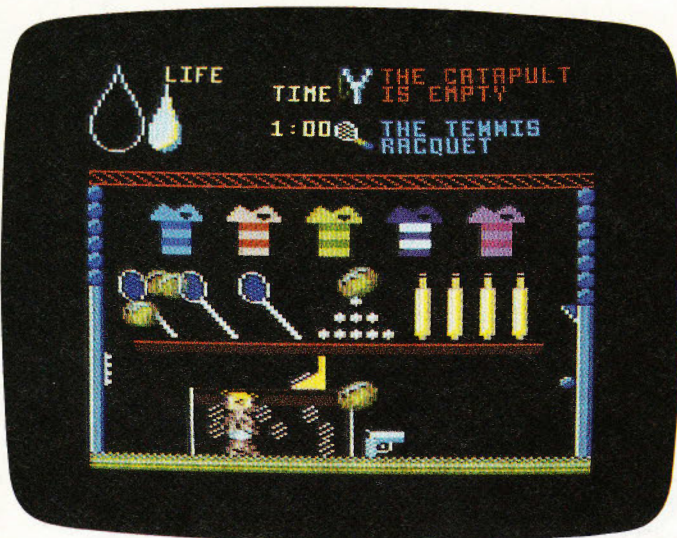




drücken Sie dann den Aktion-Knopf.
Hierauf müssen Sie das Loch (HOLE) aufnehmen, sich nach rechts bewegen und die Handtasche vor dem Krokodilmaul abstellen.
Zusammen mit dem Loch (HOLE) und dem Goldfischglas (GOLD FISH BOWL) ist es nun Ihre Aufgabe, den Bildschirm mit dem Wunschbrunnen (WISHING WELL) zu suchen. Am linken Rand dieses Raumes müssen Sie ebenfalls den Aktion-Knopf betätigen. Anschließend begeben Sie sich in den linken Bildschirm und nehmen dabei den Skelettschlüssel (SKELETON KEY) mit. Die Krokodile bewegt sich hier immer dann nicht, wenn Sie das Goldfischglas (GOLD FISH BOWL) bei sich haben. Jetzt müssen Sie wieder zum Strand zurückgehen, dort ins Meer eintauchen und den Schrank berühren. Hierauf öffnet sich die Tür und Sie können den Spinat (SPINACH) in Ihren Besitz aufnehmen.
Lassen Sie aber den Spinat später in einem Raum, den Sie leicht wiederfinden bzw. leicht erreichen können zurück.
Gehen Sie jetzt wieder in den Raum, in welchem sich das Strandgemälde befindet und nehmen Sie die gefüllte Schale (STUFFING BOWL) vom Tisch; gehen Sie darauf mit dieser Schale in den Raum mit dem großen Vogel (Saurier). Nehmen Sie hier das Ei auf, aber achten Sie darauf, daß es Ihnen nicht in die Schale fällt! Mit dem Ei und dem Spinat müssen Sie sich jetzt auf den Weg zu dem Raum machen, in dem sich das Schild "OLD FAITHFUL" befindet.
Wenn Sie hier angekommen sind, müssen Sie die Liane berühren und schnell in den nun entstandenen Wasserfall springen. Auf dem Weg nach oben sollten Sie in das Nest hüpfen und dort den Spinat gegen Pfeile und einen Bogen (BOW AND ARROWS) eintauschen. Hierauf müssen Sie aus dem Nest springen und das Ei darin zurücklassen. Mit den Pfeilen und dem Bogen müssen Sie sich jetzt in den Bildschirmteil bewegen, in dem sich Wilma befindet. Den wilden Eingeborenen können Sie nun mit einem Pfeil erschießen.

Nun müssen Sie Ihre Reise wieder zum Wunschbrunnen-Raum (WISHING WELL) fortsetzen und genau in der Mitte des Brunnenrandes den Aktion-Knopf drücken, wodurch Wally (den Sie ja steuern) auf den Grund des Brunnens fällt. Hier angekommen, gilt es, die Flasche (BOTTLE) aufzunehmen und darauf am rechten Brunnenrand wieder herauszuklettern (dabei muß der Aktion-Knopf niedergedrückt bleiben). Mit der zuvor erworbenen Flasche und dem Korkenzieher (CORKSCREW) müssen Sie jetzt zurück in den Krokodil-Raum gehen. Dadurch, daß Sie die Handtasche (HANDBAG) verwenden, können Sie mit dem Korkenzieher und der Flasche (jeweils einzeln, also Handtasche und Korkenzieher bzw. Handtasche und Flasche) an dem Krokodil vorbeikommen. Hierauf müssen Sie sich mit diesen beiden Gegenständen über die Kokusnuß stellen und abermals den Aktion-Knopf betätigen. Nehmen Sie danach die gefüllte Ölflasche (BOTTLE OF OIL) und die stumpfe Axt (BLUNT AXE) zu dem Bildschirm mit, in dem das Auto steht. Schärfen Sie nun dadurch die







Axt am Vorderrad des Wagens, daß Sie den Aktion-Knopf niederdrücken. Gehen Sie anschließend mit der jetzt scharfen Axt zu Wilma zurück, um sie durch mehrmaliges Drücken des Aktion-Knopf endlich aus ihrer unglücklichen Situation zu befreien. Damit haben Sie bereits den ersten Teil des Programms gelöst!

Herbert's Rettung:

Zuerst müssen Sie sich die Zündhölzer (DEAUX STICKS) aus dem Bildschirm, in dem sich das Krokodil befindet, besorgen. Im Amboß-Raum können diese dann dazu benutzt werden, ein Feuer anzuzünden. Nachdem Sie das Feuer zum Brennen gebracht haben, müssen Sie zum Wunschbrunnen (WISHING WELL) gehen und dort den Blasebalg (BELLOWS) vom Brunnenrand aufnehmen. Wenn Sie jetzt über dem Feuer den Aktion-Knopf niederdrücken, erhalten Sie dadurch heiße Asche (HOT ASHES).

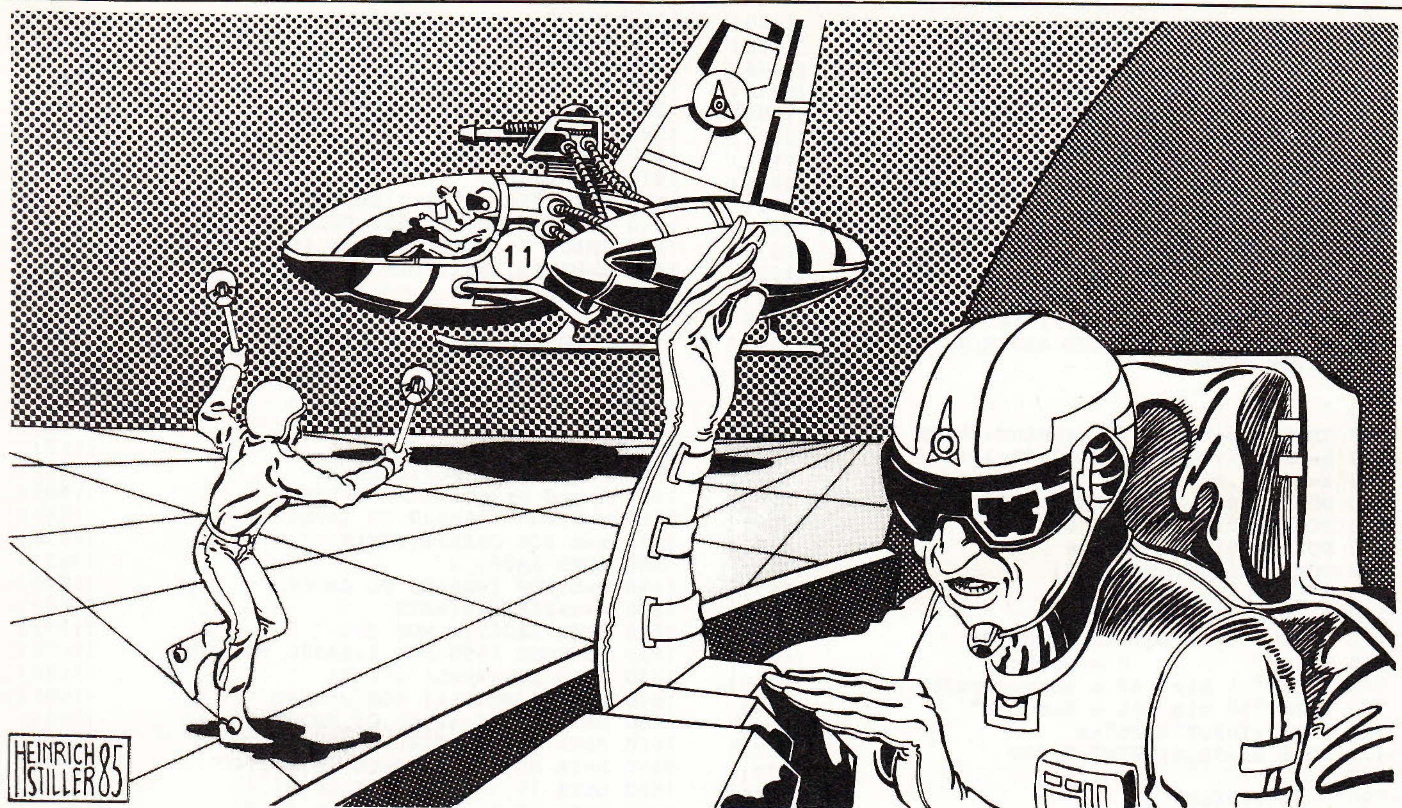
Mit dieser Asche müssen Sie sich jetzt zum Totentempel-Bildschirm begeben und den Aktion-Knopf drücken, wenn Sie bei dem weißen Mann angekommen sind. Besitzen Sie den Blasebalg, dann wird sich die Wolke, nachdem Sie sich rechts von ihr hinbegeben haben, aus dem linken Bildrand hinausbewegen. Folgen Sie der Wolke, bis diese die Hütte erreicht hat und sammeln Sie dort die leeren Muscheln (EMPTY SHELLS) auf. Nehmen Sie diese Muscheln mit in den Brunnen und drücken Sie, auf dem Brunnengrund angekommen, dann in dem Moment auf den Aktion-Knopf, wenn der Wassertropfen auftrifft.

Begeben Sie sich jetzt wieder nach oben (siehe dazu die Beschreibung bei Wilmas Rettung) und besorgen Sie sich die Plastik-Kanne (BILLY CAN). Gehen Sie damit zum Geyser, berühren Sie die Liane und rennen Sie dann zum Wasserfall, um die Kanne zu füllen. Suchen Sie hierauf die Schwimmflügel (FLIP FLOPS) und nehmen Sie diese zusammen mit der gefüllten Kanne mit zum Strand.

Wenn Sie die Krabbe erreicht haben, müssen Sie den Aktion-Knopf drücken und sich anschließend mit der Krabbenschere und der vollen Muschel von rechts zu dem Raum begeben, in dem der arme Herbert schon entsetzlich schmort. Halten Sie hier den Aktion-Knopf länger niedergedrückt, dann ist es Ihnen möglich, unversehrt an dem Löwen vorbeizukommen. Wenn Sie dann an der Mitte des Kessels angekommen sind, können Sie Herbert jetzt ganz einfach aus dem Kessel retten, indem Sie erneut den Aktion-Knopf niederdrücken.

Wenn Sie sich jetzt noch zum Strand begeben, verlassen Sie automatisch per Schiff diese verwunschene, aber doch äußerst interessante Insel und erreichen darüber hinaus noch 100 % der Lösung.

(Markus und Andreas Pisters)



ELITE: Commander nach Wunsch



"Commander Thomas ... derzeitiges System: Geisgeza ... Hyperraumziel: Arexe ... Treibstoff: 7.0 Lichtjahre ... **Konto: 65 Mio. Cr.** ... Strafregister: sauber ... **Einstufung:** Elite ... Ausrüstung: Zusatz-Laderaum ... E.C.M.-System ... **E.C.M.-Störmechanismus** ... Raumgreifer ... Rettungskapsel ... Energiebombe ... Zusatz-Energieakku ... Lande-Computer ... galaktischer Hyperraumantrieb ... **Tarnmechanismus** ... Militärlaser vorne ... Militärlaser hinten ... Militärlaser links ... Militärlaser rechts"

Dies ist meine aktuelle Statusseite im Spiel ELITE, die ich **nach ganzen zwei Tagen** erreichte! Leider kann ich Ihnen jedoch keinen Planeten verraten, auf dem u.a. eine solche "wunderbare Credit-Vermehrung" stattfindet. Aber vielleicht genügt Ihnen vorerst das untenstehende Programm: Das Programm macht nichts anderes, als auf Kassette einen Commander zu erzeugen, der anschließend vom ELITE-Programm wieder geladen werden kann. Dieser Commander wird ausschließlich im Dialog mit dem User zusammengestellt, so daß das Programm für jeden, der mit der ELITE-Terminologie vertraut ist, ohne weiteres verständlich ist.

Ein Hinweis jedoch zu der Abfrage "System eingeben?": Tippen Sie "J" oder "j", so erwartet der Rechner die Eingabe der Koordinaten des gewünschten Systems, bezogen auf die jeweilige galaktische Karte, anschließend die technische Stufe. Eine direkte Eingabe des Planetennamens ist leider nicht möglich (Sonst müßte das Programm die Koordinaten sämtlicher Planeten kennen - und Sie wären eine Weile mit dem Abtippen beschäftigt!). Geben Sie jedoch "N" oder "n" ein, so wählt der Rechner die Koordinaten eines Planeten in der Mitte der jeweiligen Galaxie.

Natürlich sollen Sie das Programm nicht dazu benutzen, sich sofort zum "Elite-Kämpfer" zu befördern (es käme ja auch niemand auf die Idee, beim Würfelspiel die Figuren gleich ins Ziel zu setzen). Es eignet sich aber hervorragend dazu, schon einmal auszuprobieren, ob z.B. an den Seiten Laser notwendig sind oder nicht. Bei sinnvoller Anwendung dieses Programms bleibt ELITE ein Spiel, das so schnell nicht langweilig wird. (Thomas Kochmann)

```
10 ' Commander fuer ELITE                          [1544]
20 '                                               [117]
30 ' Vorbereitung                                  [643]
40 MEMORY &9FFF                                    [134]
50 FOR i=&A000 TO &A067:POKE i,0:NEXT              [1482]
60 '                                               [117]
70 ' Parameter fuer Galaxien                       [1699]
80 RESTORE 130:DIM par(7,8)                        [1739]
90 FOR gal=1 TO 8                                  [640]
100 FOR i=0 TO 8:READ x$                           [1428]
110 par(gal-1,i)=VAL("&"+x$)                       [1464]
120 NEXT:NEXT                                      [1022]
130 DATA 4A,5A,48,02,53,B7,79,85,09                [2138]
140 DATA 94,B4,90,04,A6,6F,83,84,06                [1151]
150 DATA 29,69,21,08,4D,DE,78,82,0C                [979]
160 DATA 52,D2,42,10,9A,BD,7A,86,07                [911]
170 DATA A4,A5,84,20,35,7B,90,80,05                [1360]
180 DATA 49,4B,09,40,6A,F6,75,82,07                [1646]
190 DATA 92,96,12,80,D4,ED,7C,7F,09                [1284]
200 DATA 25,2D,24,01,A9,DB,7B,80,07                [1957]
210 '                                              [117]
220 ' titel                                        [643]
230 MODE 1:LOCATE 8,2                              [1569]
240 PRINT"** Commander fuer ELITE **"              [2948]
250 LOCATE 1,5                                     [623]
260 '                                              [117]
270 ' Name eingeben                                [366]
280 INPUT"Name des Commanders";a$                  [2296]
290 PRINT:PRINT:a$=UPPER$(LEFT$(a$,10))            [2682]
300 FOR i=1 TO LEN(a$)                             [791]
310 POKE &9FFF+i,ASC(MID$(a$,i,1))                 [1507]
320 NEXT:POKE &9FFF+i,0                            [1034]
330 '                                              [117]
340 ' Derzeitiges System                           [1490]
350 PRINT"Derzeitiges System:":PRINT               [3922]
360 INPUT"Galaxie Nr.";x:PRINT                     [1890]
370 x=x-1:x=x MOD 8:POKE &A011,x                   [2261]
380 FOR i=0 TO 5                                   [480]
390 POKE &A03C+i,par(x,i):NEXT                     [1594]
400 POKE &A043,par(x,6)                            [693]
410 POKE &A045,par(x,7)                            [763]
420 POKE &A048,par(x,8)                            [955]
430 INPUT"System eingeben (J/N)";a$                [3346]
440 IF UPPER$(a$)<>"J"THEN 540                     [1796]
```

```
450 PRINT:PRINT"x-Koordinate:  ";          [1360]
460 PRINT"links: 0  rechts: 255"           [1625]
470 PRINT"y-Koordinate:  ";                [2046]
480 PRINT"oben:  0  unten:  255"           [1848]
490 PRINT:INPUT"x-Koordinate";x            [1574]
500 INPUT"y-Koordinate";y                  [2384]
510 POKE &A043,x:POKE &A045,y              [1142]
520 PRINT:INPUT"Technische Stufe";x        [2818]
530 POKE &A048,MIN(x-1,255)                [619]
540 POKE &A065,&FF                         [448]
550 PRINT:PRINT                            [743]
560 '                                      [117]
570 ' Treibstoff                           [734]
580 INPUT"Treibstoff (Lj.)";x              [1125]
590 x=MIN(10*x,255):POKE &A02E,x           [3311]
600 PRINT:PRINT                            [743]
610 '                                      [117]
620 ' Konto                                [395]
630 INPUT"Konto (Cr)";x:PRINT:PRINT        [2465]
640 a=INT(x/1000):b=(x-1000*a)*10          [2793]
650 a=MIN(a,65535)                         [1817]
660 POKE &A012,a-256*INT(a/256)            [1212]
670 POKE &A013,INT(a/256)                  [755]
680 POKE &A014,b MOD 256                   [659]
690 POKE &A015,INT(b/256)                  [807]
700 '                                      [117]
710 ' Strafregister                        [1088]
720 PRINT"Strafregister:":PRINT            [1753]
730 PRINT"        0 = Sauber"              [1071]
740 PRINT" 1 bis   49 = Vorbestraft"       [2919]
750 PRINT"50 bis 255 = Gesucht"            [621]
760 PRINT:INPUT"Wert";x                    [806]
770 POKE &A00B,x:PRINT:PRINT               [2070]
780 '                                      [117]
790 ' Einstufung                           [1846]
800 PRINT"Einstufung:":PRINT               [2018]
810 PRINT"       0 bis     2303 = ";       [2261]
820 PRINT"Harmlos"                         [959]
830 PRINT"    2304 bis     4351 = ";       [1597]
840 PRINT"Relativ Harmlos"                 [2141]
850 PRINT"    4352 bis     8447 = ";       [2698]
860 PRINT"Schwach"                         [871]
870 PRINT"    8448 bis    16639 = ";       [1826]
880 PRINT"Durchschnitt"                    [1652]
890 PRINT"   16640 bis    33023 = ";       [1335]
900 PRINT"Ueberdurchschnitt"               [2747]
910 PRINT"   33024 bis   196607 = ";       [1204]
920 PRINT"Kompetent"                       [1521]
930 PRINT" 196608 bis   720895 = ";        [2007]
940 PRINT"Gefaehrlich"                     [1557]
950 PRINT" 720896 bis 1703935 = ";         [1424]
960 PRINT"Toedlich"                        [1406]
970 PRINT"1703936 bis ....... = ";         [2217]
980 PRINT"Elite"                           [747]
990 PRINT:INPUT"Wert";x                    [806]
1000 a=INT(x/65536):x=x-65536*a            [2429]
1010 b=INT(x/256):c=x-256*b                [1795]
1020 POKE &A00C,c:POKE &A00D,b             [1479]
1030 POKE &A00E,a:PRINT:PRINT              [2372]
1040 '                                     [117]
1050 ' Ausruestung                         [1063]
1060 PRINT"Ausruestung:":PRINT             [1550]
1070 INPUT"Raketen: Anzahl";x:PRINT        [3054]
1080 POKE &A02F,x MOD 256                  [1228]
1090 RESTORE 1140:FOR n=1 TO 8             [1168]
1100 READ a$,i:PRINT a$;" (J/N)";          [701]
1110 PRINT TAB(26):INPUT a$                [1821]
1120 x=0:IF UPPER$(a$)="J" THEN x=&FF      [2760]
1130 POKE i,x:NEXT                         [696]
1140 DATA Zusatz-Laderaum,&A030            [2797]
1150 DATA E.C.M.-System,&A031              [441]
1160 DATA Raumgreifer,&A034                [1095]
1170 DATA Rettungskapsel,&A035             [911]
1180 DATA Energiebombe,&A036               [840]
1190 DATA Zusatz-Energieakku,&A037         [2284]
1200 DATA Lande-Computer,&A038             [1442]
1210 DATA Gal. Hyperraumantr.,&A039        [1822]
1220 PRINT"Tarnmechanismus (J/N)";         [3231]
1230 PRINT TAB(26):INPUT a$                [1821]
1240 x=0:IF UPPER$(a$)="J"THEN x=&40       [2562]
1250 PRINT"E.C.M.-Stoermech. (J/N)";       [3535]
1260 PRINT TAB(26):INPUT a$                [1821]
1270 IF UPPER$(a$)="J"THEN x=x OR &80      [1720]
1280 POKE &A05C,x                          [770]
1290 PRINT:PRINT"Laser:  0=keiner"         [3090]
1300 PRINT TAB(9)"1=Impuls-Laser"          [639]
1310 PRINT TAB(9)"2=Strahlen-Laser"        [2015]
1320 PRINT TAB(9)"3=Militaer-Laser"        [2286]
1330 PRINT TAB(9)"4=Asteroiden-Laser"      [1315]
1340 PRINT                                 [361]
1350 INPUT"vorne ";x:POKE &A029,x          [1097]
1360 INPUT"hinten";x:POKE &A02A,x          [2120]
1370 INPUT"links ";x:POKE &A02B,x          [1463]
1380 INPUT"rechts";x:POKE &A02C,x          [2493]
1390 PRINT:PRINT                           [743]
1400 '                                     [117]
1410 ' Inventar                            [574]
1420 PRINT"Inventar:":PRINT                [1253]
1430 a=&A017:GOSUB 1850:PRINT              [2578]
1440 INPUT"Freier Laderaum (t)";x          [3128]
1450 x=MIN(x,255):POKE &A028,x             [1716]
1460 PRINT:PRINT                           [743]
1470 '                                     [117]
1480 ' Lagerbestand im System              [1500]
1490 PRINT"Lagerbestand im"                [1543]
1500 PRINT"derzeitigen System:"            [2050]
1510 PRINT:a=&A04A:GOSUB 1850              [1336]
1520 PRINT:PRINT                           [743]
1530 '                                     [117]
1540 ' Bytes verschluesseln                [1462]
1550 ' und Pruefsummen bilden              [1969]
1560 x=0:FOR i=&A000 TO &A065              [1041]
1570 x=x XOR PEEK(i):NEXT                  [1936]
1580 POKE &A066,x                          [365]
1590 x=0:FOR i=&A000 TO &A066              [1040]
1600 x=x+PEEK(i):NEXT                      [2642]
1610 POKE &A067,x MOD 256                  [1232]
1620 RESTORE 1650:FOR i=&A000 TO &A067     [1878]
1630 READ x$:x=VAL("&"+x$)                 [1680]
1640 POKE i,PEEK(i) XOR x:NEXT             [1001]
1650 DATA 21,16,4F,CD,C3,6B,21,20          [965]
1660 DATA 10,11,73,4F,01,09,00,ED          [1695]
1670 DATA B0,11,F1,69,CD,4D,64,76          [1119]
1680 DATA 76,76,76,76,76,CD,7B,6B          [1898]
1690 DATA 32,7C,4F,CD,86,6B,32,7D          [1926]
1700 DATA 4F,CD,6B,6B,3E,02,CD,CA          [1458]
1710 DATA 6B,CD,A8,56,CD,BA,6B,CD          [2016]
1720 DATA 50,CE,FB,CD,6B,6B,3E,97          [475]
1730 DATA CD,CA,6B,21,01,0C,22,0F          [1458]
1740 DATA 4F,3E,04,CD,26,63,CD,57          [1301]
1750 DATA 68,D6,15,E6,0F,5F,32,67          [1281]
1760 DATA 6B,3A,28,4F,57,32,68,6B          [1987]
1770 DATA 3A,29,4F,6F,32,69,6B,7D          [1572]
1780 '                                     [117]
1790 ' Commander abspeichern               [1480]
1800 CALL &BB03                            [396]
1810 SAVE"",b,&A000,&68                    [956]
1820 PRINT:PRINT:END                       [1018]
1830 '                                     [117]
1840 'Unterprogramme                       [916]
1850 INPUT"Nahrung (t)       ";x           [1207]
1860 GOSUB 2180                            [855]
1870 INPUT"Textilien (t)     ";x           [2107]
1880 GOSUB 2180                            [855]
1890 INPUT"Spaltstoffe (t)";x              [1381]
1900 GOSUB 2180                            [855]
1910 INPUT"Sklaven (t)       ";x           [902]
1920 GOSUB 2180                            [855]
1930 INPUT"Spirituosen (t)";x              [1492]
1940 GOSUB 2180                            [855]
1950 INPUT"Luxusgueter (t)";x              [1796]
1960 GOSUB 2180                            [855]
1970 INPUT"Rauschgifte (t)";x              [1857]
1980 GOSUB 2180                            [855]
1990 INPUT"Computer (t)      ";x           [1021]
2000 GOSUB 2180                            [855]
2010 INPUT"Maschinen (t)     ";x           [1093]
2020 GOSUB 2180                            [855]
2030 INPUT"Legierungen (t)";x              [1119]
2040 GOSUB 2180                            [855]
2050 INPUT"Waffen (t)        ";x           [1855]
2060 GOSUB 2180                            [855]
2070 INPUT"Felle (t)         ";x           [1342]
2080 GOSUB 2180                            [855]
2090 INPUT"Mineralien (t) ";x              [2006]
2100 GOSUB 2180                            [855]
2110 INPUT"Gold (kg)         ";x           [1324]
2120 GOSUB 2180                            [855]
2130 INPUT"Platin (kg)       ";x           [1983]
2140 GOSUB 2180                            [855]
2150 INPUT"Edelsteine (g) ";x              [1764]
2160 GOSUB 2180                            [855]
2170 RETURN                                [555]
2180 x=MIN(x,255):POKE a,x                 [1473]
2190 a=a+1:RETURN                          [1642]
2200 '                                     [117]
2210 ' Die Anmerkungen koennen             [2706]
2220 ' weggelassen werden                  [1643]
```

Mühle



Das Programm "Mühle" ist die gelungene Umsetzung des uralten Brettspiels für die Rechner der CPC-Reihe.
Die Spielregeln entsprechen weitestgehend denen des Originals. Folgende Besonderheiten müssen jedoch beim Spiel beachtet werden:

1. Es darf kein Stein aus einer Mühle genommen werden, auch dann nicht, wenn man sonst keinen Stein bewegen kann. In diesem Fall muß man aussetzen und gibt PE [ENTER] ein.
2. Es darf nur auf miteinander verbundene Felder gezogen werden. Beispielsweise A1 nach D1 oder A4. Bei Falscheingaben gibt der Rechner eine Fehlermeldung aus.
3. Bleiben einem Spieler nur noch drei Steine (Minimum, um eine Mühle zu bilden), so darf er mit seinen Steinen wahllos auf jedes beliebige LEERE(!!) Feld springen.
4. Beim Setzen geben Sie einfach die Position des Steins an (z.B. A1 [ENTER]).
5. Zum Nehmen geben Sie einfach die Position des Steins an (z.B. A1 [ENTER]).
6. Zum Verschieben eines Steins geben Sie zuerst die Position an, wo der Stein steht und danach die Position, wo der Stein hin soll (z.B. Stein von A1 nach D1 verschieben = A1D1 [ENTER]).
7. Wurde einer der Spieler so eingeklemmt, daß er keinen Stein mehr verschieben kann, so gibt er ein FEST [ENTER] ein und hat damit verloren.

Als zusätzliches Feature wurde die Möglichkeit zum Abspeichern des Spielstandes eingebaut. Damit kann man eine angefangene Partie unterbrechen und zu einem späteren Zeitpunkt fortsetzen.
Um ein Spiel zu speichern, geben Sie ein: ENDE [ENTER].
Daraufhin fragte der Rechner: NOCH EIN SPIEL (J/N). Geben Sie N ein und beantworten Sie die Frage: SPIELDATEN ABSPEICHRN (J/N) mit J.
Alles weitere erledigt der Rechner automatisch.

(R. Backes)

```
1 REM ---------------------- Programm ´T [3730]
ITELBILD´ ---------------------------
2 FOR Z=0 TO 15:INK Z,0:NEXT:BORDER 0:MEMO [6136]
RY &A1FF:MODE 0:POKE &B939,&C9
3 PLOT 200,320,1:DRAW 440,320:DRAW 560,80: [9586]
DRAW 80,80:DRAW 200,320:PLOT 230,290:DRAW
410,290:PLOT 410,290:DRAW 470,140:DRAW 170
,140:DRAW 230,290:PLOT 260,260:DRAW 380,26
0:PLOT 380,260:PLOT 380,260
4 DRAW 410,185:DRAW 230,185:DRAW 260,260:P [7768]
LOT 320,260:DRAW 320,79:PLOT 320,259:DRAW
320,319:PLOT 245,229:DRAW 155,229:PLOT 395
,229:PLOT 322,253:PLOT 322,255:PLOT 322,25
7,0:DRAW 322,217:DRAW 322,189
5 PLOT 322,189,0:PLOT 322,187,1:PLOT 394,2 [9466]
25:DRAW 488,225:PLOT 486,227:PLOT 562,79:D
RAW 562,63:DRAW 562,55:DRAW 82,55:DRAW 82,
83:PLOT 152,59,2:DRAW 152,75:DRAW 168,75:D
RAW 152,75:PLOT 168,75:DRAW 168,67
6 DRAW 160,67,2:DRAW 168,59:PLOT 196,63:DR [9441]
AW 196,71:PLOT 200,75:DRAW 208,75:PLOT 212
,71:DRAW 212,63:PLOT 208,59:DRAW 200,59:PL
OT 236,59:DRAW 236,75:DRAW 248,75:PLOT 252
,71:PLOT 248,67:DRAW 240,67
7 PLOT 248,67,2:PLOT 252,63:PLOT 248,59:DR [11542]
AW 240,59:PLOT 272,59:DRAW 284,59:PLOT 288
,63:PLOT 284,67:DRAW 276,67:PLOT 272,71:PL
OT 276,75:DRAW 288,75:PLOT 308,63:DRAW 308
,71:PLOT 312,75:DRAW 320,75
8 PLOT 324,71:DRAW 324,63:PLOT 320,59:DRAW [12152]
 312,59:PLOT 344,59:DRAW 344,75:DRAW 356,7
5:PLOT 356,67:DRAW 348,67:PLOT 384,59:DRAW
 384,75:DRAW 372,75:DRAW 396,75:PLOT 408,6
7:DRAW 416,67:PLOT 448,59
9 DRAW 440,59:PLOT 436,63:DRAW 436,71:PLOT [9650]
 440,75:DRAW 448,75:PLOT 464,59:DRAW 464,7
5:DRAW 472,75:PLOT 476,71:PLOT 472,67:DRAW
 468,67:PLOT 500,59:DRAW 492,59:PLOT 488,6
3:DRAW 488,71:PLOT 492,75
10 DRAW 500,75                             [413]
11 LOCATE 8,4:PEN 11:PRINT"MUEHLE":LOCATE  [10018]
10,11:PEN 12:PRINT"by":PEN 4:LOCATE 4,23:P
RINT CHR$(164):PEN 11:LOCATE 6,23:PRINT"RO
LF  BACKES":PEN 3:LOCATE 6,25:PRINT"IS LOA
DING"
12 LOCATE 7,11:PEN 2:PRINT CHR$(231):LOCAT [7523]
E 8,14:PEN 13:PRINT CHR$(231):LOCATE 9,9:P
RINT CHR$(231):LOCATE 15,17:PEN 2:PRINT CH
R$(231)
13 FOR adr=&A200 TO &A2E1:READ a$          [1803]
14 w=VAL("&"+A$):s=s+w:POKE adr,w:NEXT     [2802]
15 MOVE 320,190:CALL &A200,14:MOVE 330,100 [7979]
:CALL &A200,14:MOVE 330,270:CALL &A200,14:
MOVE 330,299:CALL &A200,14:MOVE 200,70:CAL
L &A200,12:MOVE 302,230:CALL &A200,14
16 INK 1,24:INK 2,26:INK 3,6:INK 4,21:INK  [3700]
6,11:INK 11,17:INK 12,1:INK 14,13
17 POKE &B939,&F3:RUN"!"                   [1036]
18 DATA DF,04,A2,C9,07,A2,FE,7B            [605]
19 DATA CD,F6,17,F5,CD,FC,15,CD            [2366]
20 DATA 27,18,CD,86,0C,C1,B8,C8            [1312]
21 DATA 32,DF,A2,CD,1A,16,CD,FF            [1001]
22 DATA 16,D2,95,A2,E5,FD,E1,D5            [1586]
23 DATA DD,E1,CD,A9,0B,3A,DF,A2            [1770]
24 DATA 47,FD,23,CD,2D,0C,3A,34            [1245]
25 DATA B3,FD,95,38,05,78,AE,A1            [1772]
26 DATA 28,EF,FD,2B,CD,13,0C,51            [1057]
27 DATA 59,E5,C5,CB,01,DC,05,0C            [1913]
28 DATA 78,42,AE,A1,57,20,0A,B0            [1000]
29 DATA 28,07,DD,2B,DF,A5,A2,DD            [1253]
30 DATA 23,C1,E1,E5,C5,CB,09,DC            [1187]
31 DATA F9,0B,78,43,AE,A1,5F,20            [1459]
32 DATA 0A,B0,28,07,DD,23,DF,A5            [1957]
33 DATA A2,DD,2B,C1,E1,3A,38,B3            [896]
34 DATA AE,A1,AE,77,3A,36,B3,FD            [1523]
35 DATA 95,30,0A,FD,2B,CD,13,0C            [1178]
36 DATA 78,AE,A1,28,B4,2A,E0,A2            [1304]
37 DATA 7D,B4,C8,2B,22,E0,A2,DF            [1792]
38 DATA CD,A2,C3,1E,A2,A8,A2,FD            [1736]
39 DATA D5,2A,89,AE,01,03,00,CD            [1769]
40 DATA 18,F6,38,17,EB,22,89,AE            [720]
41 DATA DD,E5,D1,FD,E5,C1,2B,72            [1370]
42 DATA 2B,73,2B,71,2A,E0,A2,23            [1333]
43 DATA 22,E0,A2,D1,C9,D0,A2,FD            [1475]
44 DATA 2A,89,AE,2B,56,2B,5E,2B            [829]
45 DATA 22,89,AE,6E,26,00,C9,00            [1510]
46 DATA 00,00                              [440]
```

```
1 POKE &BDEE
,&C9:MEMORY &AB5E:DEFINT A-Z:G [3265]
OSUB 318:GOSUB 224
2 REM---------------------- Unterprogramm [3860]
 ´wer beginnt´--------------------
3 com=PEEK(&AB7B):IF mo=0 THEN 5 ELSE IF m [3457]
o=2 THEN mo=0 ELSE IF mo=3 THEN mo=1
4 gew1=PEEK(&AB7C):gew2=PEEK(&AB7D):sp=mo: [3477]
GOTO 6
5 sp=INT(RND*2):mo=sp:POKE &AB7E,mo        [1124]
6 LOCATE#1,5,2:IF sp=0 THEN PRINT#1,TAB((1 [7131]
7-LEN(sp1$))/2+1)sp1$;"- BEGINNT"ELSE PRI
NT#1,TAB((17-LEN(sp2$))/2+1)sp2$;"- begin
nt"
7 FOR za=1 TO 1500:NEXT:CLS#1:IF mo=0 THEN [5318]
 POKE &AB7E,3 ELSE POKE &AB7E,2
8 REM --------------- ´UNTERPROGRAMM ´zieh [4239]
en oder setzen´--------------------
9 POKE &B939,&F3:IF PEEK(&DBBD)/PEEK(&D93D [2394]
)=16 THEN 51
10 POKE &B08B,&8B:POKE &B08C,&AE:IF sp=0 T [2281]
HEN 12 ELSE 13
11 REM --------------- ´UNTERPROGRAMM ´ein [4680]
gabe setzen´ ---------------------
12 IF PEEK(&AB7B)=1 THEN 128 ELSE LOCATE#2 [9945]
,5,5:INPUT#2,"SETZEN:",set$:lp1$=set$:LOC
ATE#2,12,5:PRINT#2,SPC(15):LOCATE#2,12,7:P
RINT#2,LEFT$(lp1$,4):sp=1:PEN 2:GOSUB 15:G
OTO 9
13 LOCATE#3,5,5:INPUT#3,"SETZEN:",set$:lp [8363]
2$=set$:LOCATE#3,12,5:PRINT#3,SPC(15):LOCA
TE#3,12,7:PRINT#3,LEFT$(lp2$,4):sp=0:PEN 1
```

```
:GOSUB 15:GOTO 9
14 REM --------------- ´UNTERPROGRAMM ´ste [5078]
ine setzen ´-------------------------
15 FOR za=1 TO 24:IF set$<>comset$(za)THEN [5522]
 NEXT ELSE IF PEEK(coalw(za))=255 THEN 17
16 GOSUB 21:RETURN [1572]
17 LOCATE#0,spm(za),zem(za):PRINT#0,CHR$(2 [3874]
31):IF sp=1 THEN stw=2 ELSE stw=3
18 IF mo<>sp THEN st=st+1 [477]
19 LOCATE#stw,st,2:PRINT#stw," ":GOSUB 24: [4107]
RETURN
20 REM --------------- ´UNTERPROGRAMM ´fal [3729]
sche eingaben´ ---------------------
21 POKE &B939,&F3:GOSUB 318:IF sp1$="CPC 4 [10195]
64" AND com=0 THEN RETURN ELSE LOCATE#1,2,
2:WRITE#1,"! Falsche  EINGABE !":FOR zei=
1 TO 1000:NEXT:CLS#1:IF sp=0 THEN sp=1 ELS
E sp=0
22 RETURN [555]
23 REM --------------- ´UNTERPROGRAMM ´ste [5305]
ht eine muehle´ --------------------
24 GOSUB 316:FOR za=1 TO 16:IF mu(za)<>45 [3576]
THEN mz1(za)=0
25 NEXT:GOSUB 28:FOR za=1 TO 16:IF mu(za)< [2912]
>720 THEN mz2(za)=0
26 NEXT:GOSUB 29:RETURN [871]
27 REM --------------- ´UNTERPROGRAMM ´anz [4238]
eige muehle´ -----------------------
28 POKE &B939,&F3:FOR za=0 TO 16:IF mz1(za [10957]
)=0 AND mu(za)=45 THEN CLS#1:LOCATE#1,11,1
:PRINT#1,"M";CHR$(252);"HLE":LOCATE#1,10,2
:PRINT#1,sp1$:PAPER#6,2:PEN#6,1:sp=1:mz1(z
a)=1:GOTO 30:ELSE NEXT:RETURN
29 FOR za=0 TO 16:IF mz2(za)=0 AND mu(za)= [13291]
720 THEN CLS#1:LOCATE#1,11,1:PRINT#1,"M";C
HR$(252);"HLE":LOCATE#1,10,2:PRINT#1,sp2$:
PAPER#6,1:PEN#6,2:sp=0:mz2(za)=1:GOTO 30:E
LSE NEXT:RETURN
30 DI:FOR za=1 TO 2:CLS#6:LOCATE#6,1,3:PRI [11537]
NT#6,"M HLE":LOCATE#6,2,3:PRINT#6,CHR$(252
):LOCATE#6,3,1:PRINT#6,"M":LOCATE#6,3,2:PR
INT#6,CHR$(252):LOCATE#6,3,4:PRINT#6,"L":L
OCATE#6,3,5:PRINT#6,"E":FOR zei=1 TO 30:NE
XT
31 FOR s=400 TO 100 STEP-15:SOUND 1,s,5,5, [6189]
1,1:NEXT:SOUND 1,0,30,1,1:CLS#6:FOR zei=1
TO 150:NEXT:NEXT
32 CLS#1:PAPER#6,0:PEN#6,2:CLS#6:LOCATE#6, [10654]
1,3:PRINT#6,"M";CHR$(252);"HLE":LOCATE#6,3
,1:PRINT#6,"M":LOCATE#6,3,2:PRINT#6,CHR$(2
52):LOCATE#6,3,4:PRINT#6,"L":LOCATE#6,3,5:
PRINT#6,"E":EI
33 REM --------------- ´UNTERPROGRAMM ´ste [4843]
in nehmen´ ------------------------
34 POKE &B939,&F3:IF sp=1 AND PEEK(&AB7B)= [2695]
1 THEN 153
35 LOCATE#1,1,1:PRINT#1," Welchen--------- [5495]
-Stein   wollen Sie nehmen :?"
36 IF sp=0 THEN GOSUB 40:RETURN [1885]
37 LOCATE#2,5,5:INPUT#2,"NEHMEN:",ne$:CLS [9951]
#1:LOCATE#2,5,5:PRINT#2,SPC(12):PEN#2,3:LO
CATE#2,12,8:PRINT#2,LEFT$(ne$,4):PEN#2,1
38 IF ne$="PE"THEN sp=1:RETURN ELSE FOR za [11770]
=1 TO 24:IF ne$=comset$(za)AND PEEK(coalw(
za))=240 THEN GOSUB 44:LOCATE#0,spm(za),ze
m(za):PEN 3:PRINT#0,CHR$(kpm(za)):GOSUB 24
:RETURN:ELSE NEXT
39 sp=0:GOSUB 21:GOSUB 35:com=0:RETURN [1765]
40 LOCATE#3,5,5:INPUT#3,"NEHMEN:",ne$:CLS [6209]
#1:LOCATE#3,5,5:PRINT#3,SPC(12):PEN#3,0:LO
CATE#3,12,8:PRINT#3,LEFT$(ne$,4):PEN#3,2
41 IF ne$="PE"THEN sp=0:RETURN ELSE FOR za [12887]
=1 TO 24:IF ne$=comset$(za)AND PEEK(coalw(
za))=15 THEN GOSUB 44:LOCATE#0,spm(za),zem
(za):PEN 3:PRINT#0,CHR$(kpm(za)):GOSUB 24:
RETURN:ELSE NEXT
42 sp=1:GOSUB 21:GOSUB 35:RETURN [1465]
43 REM --------------- ´UNTERPROGRAMM ´ste [4556]
ht stein in muehle´ ----------------
44 ze=1:z=1:zz=0 [901]
45 IF ne$=comset$(z+zz)AND(mz1(ze)OR mz2(z [2992]
e))=1 THEN GOTO 49
46 IF zz=2 THEN zz=0 ELSE zz=zz+1:GOTO 45 [2903]
47 IF ze=16 THEN RETURN ELSE z=z+3:ze=ze+1 [3879]
:GOTO 45
48 IF sp=0 THEN sp=1 ELSE sp=0 [2049]
49 GOSUB 21:GOSUB 34:RETURN [2355]
50 REM ------------ UNTERPROGRAMM ´anzeige [5639]
 steine verschieben´ -------------
51 IF yw=1 THEN 55 [1337]
52 CLS#2:CLS#3:PEN#2,3:LOCATE#2,5,2:PRINT# [17814]
2,sp1$:PEN#2,1:LOCATE#2,4,4:PRINT#2,"von -
 nach":LOCATE#2,2,7:PRINT#2,"von:":LOCATE#
2,9,7:PRINT#2,"nach:":PEN#3,4:LOCATE#3,5,2
:PRINT#3,sp2$:PEN#3,2:LOCATE#3,4,4:PRINT#3
,"von - nach":LOCATE#3,2,7:PRINT#3,"von:"
53 GOSUB 321:LOCATE#3,9,7:PRINT#3,"nach:": [2936]
yw=1
54 REM ----------------- UNTERPROGRAMM ´st [4836]
eine verschieben´ ----------------
55 POKE &B939,&F3:st1=0:st2=0:FOR za=1 TO [6060]
24:IF PEEK(coalw(za))=15 THEN st1=st1+1 EL
SE IF PEEK(coalw(za))=240 THEN st2=st2+1
56 NEXT:IF st1<3 OR st2<3 THEN GOSUB 87 [1887]
57 IF sp=0 AND PEEK(&AB7B)=1 THEN 175 [1575]
58 IF sp=1 THEN 70 [759]
59 LOCATE#2,5,5:INPUT#2,"",vnp$:LOCATE#2, [3810]
3,5:PRINT#2,SPC(10)
60 POKE &B939,&F3:POKE &B08B,&8B:POKE &B08 [1723]
C,&AE
61 pwl$=LEFT$(vnp$,2):pwr$=RIGHT$(vnp$,2) [3280]
62 IF vnp$="ENDE"THEN POKE &AB7E,sp+2:GOSU [2504]
B 87
63 IF vnp$="FEST"THEN POKE &AB7E,sp+2:st1= [1790]
2:GOSUB 87
64 FOR za=1 TO 24:IF pwl$<>comset$(za)THEN [11989]
 NEXT ELSE IF PEEK(coalw(za))=15 THEN neh$
=pwl$:spmn=spm(za):zemn=zem(za):ea=kpm(za)
:GOTO 66
65 sp=1:GOSUB 21:GOTO 54 [1306]
66 FOR za=1 TO 24:IF pwr$<>comset$(za)THEN [7026]
 NEXT ELSE IF PEEK(coalw(za))=255 THEN sez
$=pwr$:spms=spm(za):zems=zem(za):GOTO 68
67 sp=1:GOSUB 21:GOTO 54 [1306]
68 GOSUB 81 [817]
69 PEN 3:LOCATE#0,spmn,zemn:PRINT CHR$(ea) [9041]
:PEN 2:LOCATE#0,spms,zems:PRINT#0,CHR$(231
):LOCATE#2,6,7:PEN#2,3:PRINT#2,neh$:LOCATE
#2,14,7:PRINT#2,sez$:PEN#2,1:sp=1:GOSUB 24
70 POKE &B939,&F3:CLS#1:LOCATE#3,5,5:INPUT [9527]
#3,"",vnp$:LOCATE#3,3,5:PRINT#3,SPC(10):P
OKE &B08B,&8B:POKE &B08C,&AE:pwl$=LEFT$(vn
p$,2):pwr$=RIGHT$(vnp$,2)
71 IF vnp$="ENDE"THEN POKE &AB7E,sp+2:GOSU [2504]
B 87
72 IF vnp$="FEST"THEN POKE &AB7E,sp+2:st2= [1640]
2:GOSUB 87
73 FOR za=1 TO 24:IF pwl$<>comset$(za)THEN [10778]
 NEXT ELSE IF PEEK(coalw(za))=240 THEN neh
$=pwl$:spmn=spm(za):zemn=zem(za):ea=kpm(za
):GOTO 75
74 sp=0:GOSUB 21:GOTO 54 [1032]
75 FOR za=1 TO 24:IF pwr$<>comset$(za)THEN [7005]
 NEXT ELSE IF PEEK(coalw(za))=255 THEN sez
$=pwr$:spms=spm(za):zems=zem(za):GOTO 77
76 sp=0:GOSUB 21:GOTO 54 [1032]
77 GOSUB 81 [817]
78 PEN 3:LOCATE#0,spmn,zemn:PRINT CHR$(ea) [10355]
:PEN 1:LOCATE#0,spms,zems:PRINT#0,CHR$(231
):LOCATE#3,6,7:PEN#3,4:PRINT#3,neh$:LOCATE
#3,14,7:PRINT#3,sez$:PEN#3,2
79 GOSUB 24:sp=0:GOTO 54 [1368]
80 REM --------------- UNTERPROGRAMM ´spru [6883]
ngweite pruefen´ -----------------
81 IF st1<3 OR st2<3 THEN GOSUB 87 [1251]
82 IF sp=1 AND st2<4 OR sp=0 AND st1<4 THE [2506]
N RETURN
83 FOR za=1 TO 32:IF LEFT$(spw$(za),2)=neh [8699]
$AND RIGHT$(spw$(za),2)=sez$OR LEFT$(spw$(
za),2)=sez$AND RIGHT$(spw$(za),2)=neh$THEN
 RETURN:ELSE NEXT
84 IF sp=0 THEN sp=1 ELSE sp=0 [2049]
85 GOSUB 21:GOTO 54 [2182]
86 REM ----------------- UNTERPROGRAMM ´sp [4325]
ielende ???´ ---------------------
87 POKE &B939,&F3:IF st1<3 THEN POKE &AB7D [13345]
,PEEK(&AB7D)+1:IF PEEK(&AB7B)=1 THEN PRINT
#1," COMUTER hat verloren          schade
:":ELSE PRINT#1,TAB((24-LEN(sp1$))/2+1)sp1
$,"    hat verloren"
88 IF st2<3 THEN POKE &AB7C,PEEK(&AB7C)+1: [5832]
PRINT#1,TAB((24-LEN(sp2$))/2+1)sp2$,"
 hat verloren"
89 FOR zei=1 TO 1500:NEXT:LOCATE#4,3,5:PRI [8832]
NT#4,PEEK(43900):LOCATE#4,12,5:PRINT#4,PEE
K(&AB7D):CLS#1:PRINT#1,"  Noch ein Spiel(J
/N):"
90 IF INKEY(46)=0 THEN GOSUB 93 ELSE IF IN [3048]
```

```
KEY(45)=0 THEN POKE &AB7F,1:POKE &AB60,1:E
LSE 90
91 MODE 1:IF PEEK(&AB7B)=1 THEN GOTO 256 E [8037]
LSE CALL &BB03:MODE 0:INK 0,0:BORDER 0:PEN
 7:LOCATE 2,10:PRINT"Einen Moment BITTE":R
UN
92 REM -------------- UNTERPROGRAMM ´spiel [4598]
daten abspeichern´ ----------------
93 POKE &B939,&F3:FOR zei=1 TO 200:NEXT:CA [3696]
LL &BB03:CLS#1:PRINT#1,"Spieldaten abspei
chern ?"
94 IF INKEY(45)=0 THEN CALL &BB03 ELSE IF [3567]
INKEY(46)=0 THEN 107 ELSE 94
95 pname$=s1$+"-"+s2$:POKE &AB7F,0:gew1=PE [8287]
EK(&AB7C):gew2=PEEK(&AB7D):mo=PEEK(&AB7E):
level=PEEK(&AB5F):com=PEEK(&AB7B)
96 FOR za=1 TO 24 [1348]
97 IF PEEK(coalw(za))=255 THEN stpo$(za)=" [3088]
LF":fa(za)=3
98 IF PEEK(coalw(za))=15 THEN stpo$(za)=co [3735]
mset$(za):fa(za)=2
99 IF PEEK(coalw(za))=240 THEN stpo$(za)=c [2789]
omset$(za):fa(za)=1
100 NEXT [350]
101 SPEED WRITE 1:PEN#1,1:PRINT#1,"     Cas [15933]
ette einlegen       dann ENTER dr"+CHR$(25
2)+"cken ":CALL &BB06:CLS#1:PEN#1,3:PRINT#
1,"     REC  und  PlAY        dr"+CHR$(252
)+"cken,dann ENTER ":CALL &BB06:CLS#1
102 LOCATE#1,1,2:PEN#1,1:PRINT#1,"         [8884]
  OK ?            dann:  ENTER dr"+CHR$(252
)+"cken":CALL &BB06:CLS#1:PRINT#1,"
    SAVE:":PRINT#1,TAB((24-LEN(pname$))/2+1
)pname$
103 OPENOUT"!":PRINT#9,pname$:CLOSEOUT:OPE [9935]
NOUT"!":PRINT#9,sp1$:PRINT#9,s1$:PRINT#9,s
p2$:PRINT#9,s2$:PRINT#9,gew1:PRINT#9,gew2:
PRINT#9,mo:PRINT#9,com:PRINT#9,level
104 FOR za=&AB68 TO &AB6F:wert0=PEEK(za):P [14419]
RINT#9,wert0:NEXT:FOR za=&AB72 TO &AB79:we
rt1=PEEK(za):PRINT#9,wert1:NEXT:FOR za=1 T
O 24:PRINT#9,stpo$(za):PRINT#9,fa(za):PRIN
T#9,zem(za):PRINT#9,spm(za):NEXT:FOR za=1
TO 16:PRINT#9,mz1(za):PRINT#9,mz2(za):NEXT
105 CLOSEOUT [902]
106 REM -------------------- UNTERPROGRAM [5068]
M ´ENDE´ --------------------------
107 MODE 0:CALL &BC02:INK 1,21:INK 2,15:IN [6174]
K 3,7:INK 4,18:INK 5,27:INK 0,0:BORDER 0
108 LOCATE 1,1:PEN 4:PRINT"CPC 464 - SOFTW [11847]
EARE":LOCATE 1,3:PEN 5:PRINT"P R O D U C I
 T O N":LOCATE 5,5:PEN 2:PRINT"# # # ROBSO
FT # # #":LOCATE 8,17:PEN 1:PRINT"A U F":L
OCATE 3,21:PEN 3:PRINT"Wieder ´spielen´"
109 FOR za=&AB5F TO &AB7F:POKE za,0:NEXT:C [3242]
ALL &BB06:END
110 REM --------------- UNTERPROGRAMM ´sp [4349]
ieldaten laden´ --------------------
111 POKE &B939,&F3:WINDOW SWAP 0,1 [752]
112 CLOSEIN:PEN#0,1:PRINT#0,"    Casette e [11281]
inlegen        dann ENTER druecken ":CALL &B
B06:CLS#0:PEN#0,3:PRINT#0,"     Press    P
LAY           OK ?: dann ENTER":CALL &BB06:
CLS#0
113 ON ERROR GOTO 317 [1544]
114 OPENIN"!":INPUT#9,pname$:CLOSEIN:PRINT [13369]
#0,"           LOAD:":PRINT#0,TAB((24-LEN(p
name$))/2+1)pname$:OPENIN"!":INPUT#9,sp1$:
INPUT#9,s1$:INPUT#9,sp2$:INPUT#9,s2$:INPUT
#9,gew1:INPUT#9,gew2:INPUT#9,mo:INPUT#9,co
m:INPUT#9,level
115 FOR za=&AB68 TO &AB6F:INPUT#9,wert0:PO [15062]
KE za,wert0:NEXT:FOR za=&AB72 TO &AB79:INP
UT#9,wert1:POKE za,wert1:NEXT:FOR za=1 TO
24:INPUT#9,stpo$(za):INPUT#9,fa(za):INPUT#
9,zem(za):INPUT#9,spm(za):NEXT:FOR za=1 TO
 16:INPUT#9,mz1(za):INPUT#9,mz2(za):NEXT
116 CLOSEIN [752]
117 WINDOW SWAP 1,0 [1024]
118 POKE &AB7E,mo:POKE &AB7B,com:POKE &AB5 [3278]
F,level:za=1
119 IF fa(za)>2 THEN 120 ELSE LOCATE#0,spm [4457]
(za),zem(za):PEN fa(za):PRINT#0,CHR$(231)
120 za=za+1:IF za=25 THEN GOTO 122 [2259]
121 GOTO 119 [414]
122 PEN#4,2:LOCATE#4,3,5:PRINT#4,gew1:LOCA [3690]
TE#4,12,5:PRINT#4,gew2
123 POKE &AB7C,gew1:POKE &AB7D,gew2 [1430]
124 RETURN [555]
125 st1=0:st2=0:FOR za=1 TO 24:IF PEEK(coa [5263]
lw(za))=15 THEN st1=st1+1 ELSE IF PEEK(coa
lw(za))=240 THEN st2=st2+1
126 NEXT:CLS#1:RETURN [942]
127 REM ------------- UNTERPROGRAMM ´COMU [4637]
TER setzt´ ------------------------
128 LOCATE#1,2,2:PRINT#1,"COMPUTER setzt a [4363]
uf:":xy=1:POKE &B939,&C9
129 IF PEEK(&D93B)+PEEK(&D93D)=75 AND PEEK [3329]
(&DBBD)=240 THEN 130 ELSE 136
130 FOR za=1 TO 44 [1360]
131 IF PEEK(coalw(tp8(za)))+PEEK(coalw(tp9 [8760]
(za)))=30 AND PEEK(coalw(tp11(za)))+PEEK(c
oalw(tp10(za)))=510 THEN set$=comset$(tp11
(za)):GOTO 151
132 NEXT [350]
133 FOR za=45 TO 88 [1038]
134 IF PEEK(coalw(tp9(za)))+PEEK(coalw(tp1 [10012]
1(za)))=30 AND PEEK(coalw(tp10(za)))+PEEK(
coalw(tp12(za)))=510 THEN set$=comset$(tp1
2(za)):GOTO 151
135 NEXT [350]
136 FOR za=1 TO 16:IF mu(za)=285 THEN 149 [5073]
ELSE xy=xy+3:NEXT
137 xy=1 [552]
138 IF level<2 THEN 150 ELSE FOR za=1 TO 1 [5418]
6:IF mu(za)=735 THEN 149 ELSE xy=xy+3:NEXT
139 GOSUB 315:IF level<3 THEN 150 [2346]
140 FOR za=1 TO 104:IF PEEK(coalw(tp8(za)) [9892]
)+PEEK(coalw(tp9(za)))=30 AND PEEK(coalw(t
p11(za)))+PEEK(coalw(tp12(za)))+PEEK(coalw
(tp10(za)))=765 THEN set$=comset$(tp10(za)
):GOTO 151
141 NEXT:FOR za=1 TO 104:IF PEEK(coalw(tp8 [8772]
(za)))+PEEK(coalw(tp9(za)))=480 AND PEEK(c
oalw(tp11(za)))+PEEK(coalw(tp12(za)))+PEEK
(coalw(tp10(za)))=765 THEN set$=comset$(tp
10(za)):GOTO 151
142 NEXT:IF level<4 THEN 150 ELSE FOR za=7 [13071]
7 TO 88:IF PEEK(coalw(tp8(za)))=15 AND PEE
K(coalw(tp9(za)))+PEEK(coalw(tp10(za)))+PE
EK(coalw(tp11(za)))+PEEK(coalw(tp12(za)))=
1020 THEN set$=comset$(tp9(za)):GOTO 151
143 IF PEEK(coalw(tp9(za)))=15 AND PEEK(co [11978]
alw(tp8(za)))+PEEK(coalw(tp10(za)))+PEEK(c
oalw(tp11(za)))+PEEK(coalw(tp12(za)))=1020
 THEN set$=comset$(tp8(za)):GOTO 151
144 IF PEEK(coalw(tp8(za)))+PEEK(coalw(tp9 [10658]
(za)))=30 AND PEEK(coalw(tp10(za)))+PEEK(c
oalw(tp11(za)))+PEEK(coalw(tp12(za)))=765
THEN set$=comset$(tp10(za)):GOTO 151
145 NEXT:FOR za=77 TO 88:IF PEEK(coalw(tp8 [11730]
(za)))=240 AND PEEK(coalw(tp9(za)))+PEEK(c
oalw(tp10(za)))+PEEK(coalw(tp11(za)))+PEEK
(coalw(tp12(za)))=1020 THEN set$=comset$(t
p9(za)):GOTO 151
146 IF PEEK(coalw(tp9(za)))=240 AND PEEK(c [11372]
oalw(tp8(za)))+PEEK(coalw(tp10(za)))+PEEK(
coalw(tp11(za)))+PEEK(coalw(tp12(za)))=102
0 THEN set$=comset$(tp8(za)):GOTO 151
147 IF PEEK(coalw(tp8(za)))+PEEK(coalw(tp9 [8931]
(za)))=240 AND PEEK(coalw(tp10(za)))+PEEK(
coalw(tp11(za)))+PEEK(coalw(tp12(za)))=765
 THEN set$=comset$(tp10(za)):GOTO 151
148 NEXT:GOTO 150 [969]
149 FOR zz=0 TO 2:IF PEEK(coalw(xy+zz))=25 [5721]
5 THEN set$=comset$(xy+zz):GOTO 151:ELSE N
EXT
150 za=INT(RND*24)+1:IF PEEK(coalw(za))=25 [4495]
5 THEN set$=comset$(za):ELSE GOTO 150
151 lp1$=set$:LOCATE#1,22,2:PRINT#1,set$:L [5150]
OCATE#2,12,7:PRINT#2,"     ":LOCATE#2,12,7
:PRINT#2,UPPER$(lp1$):sp=1:PEN 2:GOSUB 15:
com=0:CLS#1:GOTO 9
152 REM -------------      UNTERPROGRAMM ´co [3979]
mputer nimmt´ ---------------------
153 CLS#1:LOCATE#1,3,2:PRINT#1,"COMPUTER N [5463]
IMMT :":GOSUB 316:kn$=" ":ne$=" ":POKE &B9
39,&C9
154 FOR za=1 TO 16:IF mu(za)=720 THEN kn$= [5029]
kn$+comne$(za)
155 NEXT [350]
156 FOR lw=1 TO 3:RESTORE 308:FOR ze=1 TO [3204]
16:FOR za=1 TO 3:READ tmu1(za)
157 IF mu(ze)=tpsc(lw)AND mu(tmu1(za))=tpw [6789]
e(lw)OR mu(ze)=tpwe(lw)AND mu(tmu1(za))=tp
sc(lw)THEN 159
158 NEXT za,ze,lw:GOTO 162 [1060]
159 FOR zz=1 TO 6 STEP 2:FOR zz1=1 TO 6 ST [1790]
```

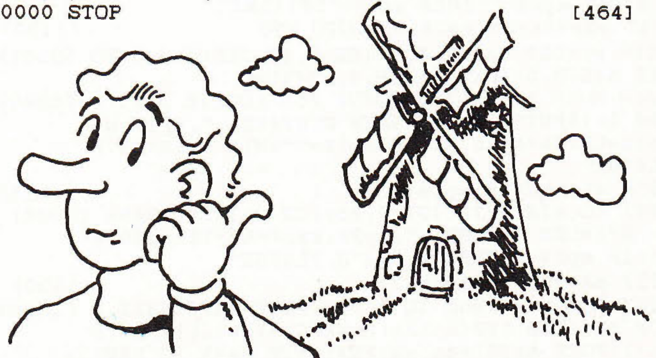
LISTINGS

```
EP 2
160 IF MID$(comne$(ze),zz,2)=MID$(comne$(t [11542]
mul(za)),zz1,2)THEN ne$=MID$(comne$(tmul(z
a)),zz1,2):FOR zz2=1 TO 24:IF ne$=comset$(
zz2)AND PEEK(coalw(zz2))=240 THEN 173 ELSE
 NEXT
161 NEXT zz1,zz:GOTO 158                  [1441]
162 xy=1:FOR za=1 TO 16:IF mu(za)=735 THEN [4422]
 171 ELSE xy=xy+3:NEXT
163 xy=1:FOR za=1 TO 16:IF mu(za)=270 THEN [4098]
 171 ELSE xy=xy+3:NEXT
164 zaw=0:xw1=0                           [1160]
165 xw=INT(RND*24)+1:POKE &B08B,&8B:POKE & [3225]
B08C,&AE
166 IF PEEK(coalw(xw))=240 THEN ne$=comset [2857]
$(xw)
167 IF INSTR(kn$,ne$)=0 THEN 173          [1541]
168 IF xw<24 THEN xw=xw+1:GOTO 166        [2345]
169 IF xw1=1 THEN ne$="PE":PRINT#1,"   Kan [9819]
n leider  keinen        Stein  nehmen !":FO
R zei=1 TO 1500:NEXT:CLS#1:sp=1:GOTO 9
170 IF zaw<3 THEN zaw=zaw+1:GOTO 165 ELSE [3822]
xw=1:xw1=1:GOTO 166
171 FOR zz=0 TO 2:IF PEEK(coalw(xy+zz))=24 [5384]
0 THEN ne$=comset$(xy+zz)ELSE NEXT:GOTO 16
4
172 IF INSTR(kn$,ne$)<>0 THEN 164         [2252]
173 LOCATE#1,20,2:PRINT#1,"";ne$:GOSUB 38 [7077]
:FOR za=1 TO 1000:NEXT:CLS#1:PEN#2,3:LOCAT
E#2,12,8:PRINT#2,LEFT$(ne$,4):PEN#2,1:GOTO
 9
174 REM -------------------- COMPUTER ´v [5269]
erschiebt´ ------------------------
175 POKE &B939,&F3:vnp$="":nz1$="":GOSUB 2 [1651]
22
176 LOCATE#1,1,1:PRINT#1,"      COMPUTER  [4021]
zieht ":POKE &B939,&C9
177 xy=1:FOR za=1 TO 16:IF mu(za)=495 THEN [5401]
 FOR zz=0 TO 2:IF PEEK(coalw(xy+zz))=15 TH
EN nz$(xy+zz)=comset$(xy+zz) ELSE NEXT
178 FOR ze=1 TO 48:IF nz$(ze)=comset$(ze)T [3314]
HEN nz1$=nz$(ze)ELSE 181
179 FOR ze1=1 TO 48:IF comset$(ze1)=nz1$TH [4642]
EN nz$(ze1)=comset$(ze1)
180 NEXT ze1                              [374]
181 NEXT ze                               [403]
182 xy=xy+3:NEXT                          [902]
183 IF st1<3 THEN 87 ELSE IF st1<>3 THEN 1 [1738]
94
184 xy=1:FOR za=1 TO 16:IF mu(za)=285 THEN [3872]
 187 ELSE xy=xy+3:NEXT
185 xy=1:FOR za=1 TO 16:IF mu(za)=735 THEN [3433]
 192 ELSE xy=xy+3:NEXT
186 GOTO 194                              [419]
187 sst$="":sst1$="":sst2$="":FOR zz=0 TO  [5226]
2:IF PEEK(coalw(xy+zz))=15 THEN sst$=sst$+
comset$(xy+zz)
188 IF PEEK(coalw(xy+zz))=255 THEN sst2$=c [3691]
omset$(xy+zz)
189 NEXT                                  [350]
190 FOR zs=1 TO 48:IF PEEK(coalw(zs))=15 A [10098]
ND LEFT$(sst$,2)<>comset$(zs)AND RIGHT$(ss
t$,2)<>comset$(zs)THEN sst1$=comset$(zs):v
np$=sst1$+sst2$:GOTO 221 ELSE NEXT
191 GOTO 194                              [419]
192 wst1$="":wst2$="":FOR zz=0 TO 2:IF PEE [6269]
K(coalw(xy+zz))=255 THEN wst2$=comset$(xy+
zz):ELSE NEXT
193 FOR zs=1 TO 48:IF PEEK(coalw(zs))=15 A [8522]
ND nz$(zs)<>comset$(zs)THEN wst1$=comset$(
zs):vnp$=wst1$+wst2$:GOTO 221 ELSE NEXT
194 xy=1:FOR za=1 TO 16:IF mu(za)=285 THEN [2367]
 195 ELSE 197
195 FOR z=0 TO 2:IF PEEK(coalw(xy+z))=255  [5621]
AND PEEK(coalw(tp13(xy+z)))=15 THEN vnp$=c
omset$(tp13(xy+z))+comset$(xy+z):GOTO 221
196 IF PEEK(coalw(xy+z))=255 AND PEEK(coal [5798]
w(tp17(xy+z)))=15 THEN vnp$=comset$(tp17(x
y+z))+comset$(xy+z):GOTO 221 ELSE NEXT
197 xy=xy+3:NEXT                          [902]
198 GOSUB 319:xy=1:FOR za=1 TO 16:IF muh1= [6702]
0 AND mu(za)=45 THEN 199 ELSE xy=xy+3:NEXT
:GOTO 200
199 FOR zz=0 TO 2:tw$=comset$(xy+zz):FOR z [10757]
s=1 TO 48:IF tw$=comset$(tp14(zs)) AND PEE
K(coalw(tp13(zs)))=255 THEN vnp$=tw$+comse
t$(tp13(zs)):GOTO 221 ELSE NEXT zs,zz
200 xy=1:FOR za=1 TO 16:IF mu(za)=735 THEN [2944]
 201 ELSE 202
201 FOR z=0 TO 2:IF PEEK(coalw(xy+z))=255  [6683]
AND PEEK(coalw(tp13(xy+z)))=15 THEN vnp$=c
omset$(tp13(xy+z))+comset$(xy+z):GOTO 221
ELSE NEXT
202 xy=xy+3:NEXT                          [902]
203 FOR za=1 TO 24:IF PEEK(coalw(tp8(za))) [12791]
+PEEK(coalw(tp9(za)))+PEEK(coalw(tp10(za))
)=45 AND PEEK(coalw(tp11(za)))=255 THEN vn
p$=comset$(tp10(za))+comset$(tp11(za)):GOT
O 221 ELSE NEXT
204 FOR za=1 TO 36:IF PEEK(coalw(tp1(za))) [12756]
+PEEK(coalw(tp2(za)))+PEEK(coalw(tp3(za)))
=45 AND PEEK(coalw(tp4(za)))+PEEK(coalw(tp
5(za)))=510 THEN vnp$=comset$(tp1(za))+com
set$(tp4(za)):GOTO 221 ELSE NEXT
205 FOR za=1 TO 12:IF PEEK(coalw(tp1(za))) [10539]
+PEEK(coalw(tp4(za)))+PEEK(coalw(tp5(za)))
=45 AND PEEK(coalw(tp2(za)))=255 THEN vnp$
=comset$(tp4(za))+comset$(tp5(za)):GOTO 22
1 ELSE NEXT
206 FOR za=1 TO 12:IF PEEK(coalw(tp2(za))) [13421]
+PEEK(coalw(tp5(za)))+PEEK(coalw(tp15(za))
)=45 AND PEEK(coalw(tp1(za)))+PEEK(coalw(t
p4(za)))=510 THEN vnp$=comset$(tp15(za))+c
omset$(tp1(za)):GOTO 221 ELSE NEXT
207 FOR za=1 TO 12:IF PEEK(coalw(tp2(za))) [13120]
+PEEK(coalw(tp4(za)))+PEEK(coalw(tp16(za))
)=45 AND PEEK(coalw(tp3(za)))+PEEK(coalw(t
p5(za)))=510 THEN vnp$=comset$(tp16(za))+c
omset$(tp3(za)):GOTO 221 ELSE NEXT
208 FOR za=21 TO 36:IF PEEK(coalw(tp1(za)) [12578]
)+PEEK(coalw(tp2(za)))+PEEK(coalw(tp3(za))
)=45 AND PEEK(coalw(tp4(za)))+PEEK(coalw(t
p5(za)))=510 AND PEEK(coalw(tp6(za-20)))=2
40 THEN vnp$=comset$(tp1(za))+comset$(tp4(
za)):GOTO 221 ELSE NEXT
209 FOR za=21 TO 36:IF PEEK(coalw(tp1(za)) [15496]
)+PEEK(coalw(tp2(za)))+PEEK(coalw(tp3(za))
)=45 AND PEEK(coalw(tp4(za)))+PEEK(coalw(t
p5(za)))=510 AND PEEK(coalw(tp7(za-20)))=2
40 THEN vnp$=comset$(tp3(za))+comset$(tp5(
za)):GOTO 221 ELSE NEXT
210 FOR za=21 TO 36                       [931]
211 IF PEEK(coalw(tp1(za)))+PEEK(coalw(tp2 [12466]
(za)))+PEEK(coalw(tp3(za)))=45 AND PEEK(co
alw(tp4(za)))+PEEK(coalw(tp5(za)))=510 AND
 PEEK(coalw(tp6(za-20)))+PEEK(coalw(tp7(za
-20)))=510 THEN vnp$=comset$(tp3(za))+coms
et$(tp5(za)):GOTO 221 ELSE NEXT
212 FOR za=21 TO 36:IF PEEK(coalw(tp1(za)) [13876]
)+PEEK(coalw(tp3(za)))+PEEK(coalw(tp6(za-2
0)))=45 AND PEEK(coalw(tp5(za)))+PEEK(coal
w(tp4(za)))=510 THEN vnp$=comset$(tp3(za))
+comset$(tp5(za)):GOTO 221 ELSE NEXT
213 FOR za=21 TO 36:IF PEEK(coalw(tp1(za)) [12638]
)+PEEK(coalw(tp3(za)))+PEEK(coalw(tp7(za-2
0)))=45 AND PEEK(coalw(tp4(za)))+PEEK(coal
w(tp5(za)))=510 THEN vnp$=comset$(tp1(za))
+comset$(tp4(za)):GOTO 221 ELSE NEXT
214 FOR za=21 TO 36:IF PEEK(coalw(tp1(za)) [11122]
)+PEEK(coalw(tp2(za)))+PEEK(coalw(tp7(za-2
0)))=45 AND PEEK(coalw(tp4(za)))+PEEK(coal
w(tp5(za)))=510 THEN vnp$=comset$(tp7(za-2
0))+comset$(tp5(za)):GOTO 221 ELSE NEXT
215 FOR zz=1 TO 48:IF PEEK(coalw(tp14(zz)) [5955]
)=255 AND PEEK(coalw(tp13(zz)))=15 THEN vn
p$=comset$(tp13(zz))+comset$(tp14(zz))ELSE
 218
216 FOR zs=1 TO 48:IF LEFT$(vnp$,2)=nz$(zs [3533]
)THEN 218
217 NEXT:GOTO 221                         [1169]
218 NEXT                                  [350]
219 FOR zz=1 TO 48:IF PEEK(coalw(tp14(zz)) [6510]
)=255 AND PEEK(coalw(tp13(zz)))=15 THEN vn
p$=comset$(tp13(zz))+comset$(tp14(zz)):GOT
O 221 ELSE NEXT
220 vnp$="ENDE":PRINT#1,"  Habe leider  ve [7671]
rloren          schade":FOR za=1 TO 1500:NE
XT:POKE &AB7D,PEEK(&AB7D)+1:GOTO 60
221 LOCATE#1,1,2:PRINT#1,"     von: ";LEFT$( [3890]
vnp$,2);"     nach: ";RIGHT$(vnp$,2):GOTO 60
222 FOR za=1 TO 48:nz$(za)="":NEXT:RETURN [2578]
223 REM ------------ UNTERPROGRAMM ´altes [4474]
oder neues spiel´ -----------------
224 POKE &B939,&F3:DIM spw$(32),spm(24),ze [7406]
m(24),kpm(24),mz1(16),mz2(16),mu(16),stpo$
(24),fa(24),coalw(48),comset$(48),comne$(1
6),nz$(48)
225 DIM tp1(52),tp2(52),tp3(52),tp4(52),tp [7589]
```

```
5(52),tp6(16),tp7(16),tp8(104),tp9(104),tp
10(104),tp11(104),tp12(104),tp13(48),tp14(
48),tp15(12),tp16(12),tp17(48)
226 SYMBOL 252,102,0,102,102,102,102,62,0: [7443]
SYMBOL 253,102,0,0,60,102,102,60,0:SYMBOL
254,102,60,102,102,102,102,60,0:SYMBOL 255
,102,0,60,6,62,102,62,0
227 SPEED KEY 255,255:st=6:mo=PEEK(&AB7E): [5893]
com=PEEK(&AB7B):level=PEEK(&AB5F)
228 s1$="":FOR za=&AB68 TO &AB70:s1$=s1$+C [3494]
HR$(PEEK(za)):NEXT:s1$=LEFT$(s1$,3)
229 s2$="":FOR za=&AB72 TO &AB7A:s2$=s2$+C [4298]
HR$(PEEK(za)):NEXT:s2$=LEFT$(s2$,3)
230 IF PEEK(&AB7F)=1 THEN 260 [1393]
231 BORDER 0:INK 0,13:INK 1,2:INK 2,21:INK [11261]
 3,0:MODE 1:WINDOW#1,1,40,1,8:WINDOW#2,1,4
0,9,18:WINDOW#3,1,40,19,22:WINDOW#4,1,40,2
3,25:PAPER#2,0:PAPER#1,2:PAPER#3,2:PAPER#4
,3:CLS#4:CLS#2:CLS#3:CLS#1
232 LOCATE#1,12,2:PEN#1,3:PRINT#1,"R  O  B [8310]
  S  O  F  T":LOCATE#1,17,4:PEN#1,0:PRINT#
1,"present´s":LOCATE#1,11,6:PEN#1,1:PRINT#
1,"M     ";CHR$(252);"     H     L     E"
233 PEN#2,1:LOCATE#2,13,4:PRINT#2,"Ich w"+ [6179]
CHR$(252)+"nsche Ihnen":LOCATE#2,15,7:PEN#
2,3:PRINT#2,"´viel  SPASS´"
234 LOCATE#3,12,2:PEN#3,1:PRINT#3,CHR$(164 [7687]
):LOCATE#3,15,2:PEN#3,1:PRINT#3,"by ROLF B
ACKES":LOCATE#3,17,4:PEN#3,3:PRINT#3,"in 1
986"
235 PEN#4,2:LOCATE#4,9,2:PRINT#4,"BITTE EI [4479]
NE TASTE DR"+CHR$(252)+"CKEN":CALL &BB06
236 REM --------------- UNTERPROGRAMM ´spi [4243]
elanleitung ´ ---------------------
237 CLS#0:CALL &BC02:BORDER 13:INK 0,0:INK [2641]
 1,15:INK 2,18:INK 3,11
238 MODE 1:GOSUB 258:LOCATE 13,7:PEN 2:PRI [12302]
NT"M";CHR$(253);"chten Sie :":LOCATE 3,11:
PRINT"Ein":LOCATE 7,11:PEN 1:PRINT"ALTES":
LOCATE 13,11:PEN 2:PRINT"Spiel weiterspiel
en ?:":LOCATE 36,11:PEN 1:PRINT"(A)":LOCAT
E 16,15:PEN 2:PRINT"oder :"
239 LOCATE 4,19:PRINT"Ein":LOCATE 9,19:PEN [8136]
 1:PRINT"NEUES":LOCATE 16,19:PEN 2:PRINT"S
piel anfangen  ?:":LOCATE 35,19:PEN 1:PRIN
T"(N)"
240 PEN 2:LOCATE 6,23:PRINT"Ihre Wahl :":L [11227]
OCATE 19,23:PEN 1:PRINT"N":LOCATE 22,23:PE
N 2:PRINT"oder":LOCATE 28,23:PEN 1:PRINT"A
":LOCATE 30,23:PEN 2:PRINT": ?"
241 IF INKEY(69)=0 THEN bab=1:CALL &BB03:G [5102]
OTO 260:ELSE IF INKEY(46)=0 THEN bab=0:POK
E &AB7C,0:POKE &AB7D,0:POKE &AB7E,0:ELSE 2
41
242 CALL &BC02:BORDER 13:INK 0,0:INK 1,15: [4808]
INK 2,18:INK 3,20:MODE 1:GOSUB 258
243 LOCATE 6,7:PEN 3:PRINT"Gegen wen m";CH [10721]
R$(253);"chten Sie spielen:":LOCATE 7,12:P
EN 2:PRINT"Mensch  gegen  Mensch :":LOCATE
 33,12:PEN 1:PRINT"(M)"
244 LOCATE 16,15:PEN 3:PRINT" oder ":LOCAT [15387]
E 7,18:PEN 2:PRINT"Mensch  gegen Computer
:":LOCATE 33,18:PEN 1:PRINT"(C)":PEN 3:LOC
ATE 5,24:PRINT"Ihre Wahl :":LOCATE 17,24:
PEN 1:PRINT"(M)":LOCATE 22,24:PEN 3:PRINT"
oder":LOCATE 28,24:PEN 1:PRINT"(C)  ?"
245 CALL &BB03:IF INKEY(38)=0 THEN CALL &B [14466]
B03:GOTO 248 ELSE IF INKEY(62)=0 THEN CALL
 &BB03:sp1$="CPC 464":s1$=LEFT$(sp1$,3):co
m=1:POKE &AB7B,1:CLS:GOSUB 258:LOCATE 4,7:
PEN 2:PRINT"IHREN NAMEN (mit ENTER best";
CHR$(255);"tigen) :"ELSE 245
246 LOCATE 10,9:INPUT;sp2$:sp2$=LEFT$(sp2$ [4835]
,8):IF sp2$=""THEN sp2$="SPIELER"
247 s2$=LEFT$(sp2$,3):GOTO 253 [1156]
248 BORDER 1:INK 3,1:INK 0,12:CLS:com=0:PO [3801]
KE &AB7B,0:POKE &AB5F,0:LEVEL=0
249 MODE 1:INK 1,0:GOSUB 258:LOCATE 3,10:P [6940]
EN 1:INPUT"NAME SPIELER SCHWARZ :",sp1$:s
p1$=LEFT$(sp1$,8):IF sp1$=""THEN sp1$="1.P
LAYER"
250 s1$=LEFT$(sp1$,3) [1286]
251 LOCATE 3,14:INK 2,26:PEN 2:INPUT"NAME [5396]
  SPIELER  WEISS :",sp2$:sp2$=LEFT$(sp2$,8
):IF sp2$=""THEN sp2$="2.PLAYER"
252 s2$=LEFT$(sp2$,3) [450]
253 FOR za=&AB68 TO &AB7A:POKE za,0:NEXT:F [12151]
OR za=1 TO LEN(sp1$):na=ASC(MID$(sp1$,za,z
a)):POKE &AB67+za,na:NEXT:FOR za=1 TO LEN(
sp2$):na=ASC(MID$(sp2$,za,za)):POKE &AB71+
za,na:NEXT
254 IF PEEK(&AB7B)=0 THEN 260 [1506]
255 CLS:IF sp1$<>"CPC 464"OR bab=1 THEN RE [2412]
TURN
256 BORDER 12:INK 0,1:INK 1,24:GOSUB 258:P [8122]
EN 1:LOCATE 3,10:PRINT"Welchen Schwierigk
eitsgrad 1-4 : ?"
257 level$=INKEY$:level=VAL(level$):POKE & [6120]
AB5F,level:IF level=0 OR level>4 THEN 257
ELSE 260
258 LOCATE 3,2:PEN 3:PRINT"# # # # #    M [7482]
";CHR$(252);" H L E      # # # # #":PEN 2:P
RINT:PRINT STRING$(40,"-"):RETURN
259 REM ------------------ UNTERPROGRAMM [3820]
´spielfeldaufbau´ -----------------
260 IF PEEK(&AB60)=1 AND PEEK(&AB7B)=1 THE [10897]
N POKE &AB60,0:MODE 0:LOCATE 2,10:PRINT"Ei
nen Moment BITTE":RUN:ELSE POKE &AB60,0:MO
DE 1:INK 0,13:INK 1,26:INK 2,0:INK 3,24:BO
RDER 1
261 WINDOW#0,1,24,1,23:WINDOW#1,1,24,24,25 [8286]
:WINDOW#2,25,40,3,10:WINDOW#3,25,40,11,18:
WINDOW#4,25,40,19,25:WINDOW#5,25,40,1,2:WI
NDOW#6,11,15,11,15
262 PAPER#0,0:PAPER#1,2:PAPER#2,2:PAPER#3, [7731]
1:PAPER#4,3:PAPER#5,3:PAPER#6,0:CLS#0:CLS#
1:CLS#2:CLS#3:CLS#4:CLS#5:CLS#6
263 PLOT 56,56,3:DRAW 56,344:DRAW 344,344: [15334]
DRAW 344,56:DRAW 56,56:PLOT 55,55:DRAW 55,
343:DRAW 343,343:DRAW 343,55:DRAW 55,55:PL
OT 104,104:DRAW 104,296:DRAW 296,296:DRAW
296,104:DRAW 104,104:PLOT 103,103:DRAW 103
,295:DRAW 295,295:DRAW 295,103:DRAW 103,10
3
264 PLOT 152,152:DRAW 152,248:DRAW 248,248 [7374]
:DRAW 248,152:DRAW 152,152:PLOT 151,151:DR
AW 151,247:DRAW 247,247:DRAW 247,151:DRAW
151,151
265 PLOT 200,56:DRAW 200,152:PLOT 199,55:D [10413]
RAW 199,153:PLOT 200,344:DRAW 200,248:PLOT
 199,343:DRAW 199,249:PLOT 56,200:DRAW 152
,200:PLOT 55,199:DRAW 151,199:PLOT 344,200
:DRAW 248,200:PLOT 343,199:DRAW 249,199
266 PEN#0,3:PEN#1,3:PEN#2,1:PEN#3,2:PEN#4, [12951]
0:PEN#5,0:PEN#6,2:zahl=1:bust=65:FOR za=4
TO 22 STEP 3:LOCATE#0,za,2:PEN 2:PRINT CHR
$(bust):LOCATE#0,1,za:PRINT zahl:zahl=zahl
+1:bust=bust+1:NEXT za:PEN 3
267 LOCATE#6,1,3:PRINT#6,"M";CHR$(252);"HL [12346]
E":LOCATE#6,3,1:PRINT#6,"M":LOCATE#6,3,2:P
RINT#6,CHR$(252):LOCATE#6,3,4:PRINT#6,"L":
LOCATE#6,3,5:PRINT#6,"E":PEN 3:LOCATE#5,4,
2:PRINT#5,"SPIELSTAND"
268 IF bab=1 THEN GOSUB 111:GOTO 269:ELSE [12348]
sp1$="":sp2$="":FOR za=&AB68 TO &AB70:sp1$
=sp1$+CHR$(PEEK(za)):NEXT:FOR za=&AB72 TO
&AB7A:sp2$=sp2$+CHR$(PEEK(za)):NEXT
269 IF sp1$=sp2$THEN sp1$="1 "+LEFT$(sp1$, [3437]
LEN(sp1$)-2):sp2$="2 "+LEFT$(sp2$,LEN(sp2$
)-2)
270 PEN#2,3:LOCATE#2,2,3:PRINT#2,sp1$:PEN# [12949]
2,1:LOCATE#2,12,4:PRINT#2,"ZUG":LOCATE#2,2
,7:PRINT#2,"LETZTE P.:":PEN#3,4:LOCATE#3,2
,3:PRINT#3,sp2$:PEN#3,2:LOCATE#3,12,4:PRIN
T#3,"ZUG":LOCATE#3,2,7:PRINT#3,"LETZTE P.:
":GOSUB 321
271 IF s1$=s2$THEN s1$=LEFT$(s1$,1)+" 1":s [3523]
2$=LEFT$(s2$,1)+" 2"
272 PEN#4,2:LOCATE#4,3,1:PRINT#4,"GEW. SPI [9194]
ELE:":LOCATE#4,2,2:PRINT#4,STRING$(14,154)
:LOCATE#4,1,3:PRINT#4,TAB(((17-LEN(s2$))/2
)+5);LEFT$(s2$,5):LOCATE#4,1,3:PRINT#4,TAB
((7-LEN(s1$))/2+1);LEFT$(s1$,5)
273 FOR za=3 TO 5:LOCATE#4,8,za:PRINT#4,CH [5378]
R$(149):LOCATE#4,9,za:PRINT#4,CHR$(149):NE
XT
274 LOCATE#4,2,6:PRINT#4,STRING$(14,154):I [5513]
F level=0 THEN 275 ELSE PEN#4,0:LOCATE#4,5
,7:PRINT#4,"LEVEL :";level
275 IF bab=1 THEN GOSUB 279:GOSUB 125:GOTO [3454]
 3
276 LOCATE#4,3,5:PEN#4,2:PRINT#4,PEEK(&AB7 [4158]
C):LOCATE#4,12,5:PRINT#4,PEEK(&AB7D)
277 FOR za=7 TO 15:LOCATE#2,za,2:PRINT#2,C [5126]
HR$(230):LOCATE#3,za,2:PRINT#3,CHR$(230):N
EXT
278 REM ------------------ UNTERPROGRAMM ´ [3769]
werte einlesen´ -------------------
```

LISTINGS

```
279 POKE &B939,&C9:FOR za=1 TO 24:READ spm [3381]
(za):READ zem(za):READ kpm(za):NEXT
280 DATA 4,4,150,13,4,158,22,4,156,7,7,150 [11923]
,13,7,159,19,7,156,10,10,150,13,10,155,16,
10,156,4,13,151,7,13,159,10,13,157,16,13,1
51,19,13,159,22,13,157,10,16,147,13,16,158
,16,16,153,7,19,147,13,19,159,19,19,153,4,
22,147,13,22,155,22,22,153
281 FOR za=1 TO 32:READ spw$(za):NEXT [3342]
282 DATA A1D1,D1D2,D1G1,B2D2,D2D3,D2F2,C3D [10003]
3,D3E3,C5D5,D5D6,D5E5,B6D6,D6D7,D6F6,A7D7,
D7G7,A1A4,A4B4,A4A7,B2B4,B4C4,B4B6,C3C4,C4
C5,E3E4,E4F4,E4E5,F2F4,F4G4,F4F6,G1G4,G4G7
283 FOR za=1 TO 48:READ coalw(za):READ com [3879]
set$(za):NEXT
284 DATA &D8F7,A1,&D908,D1,&D91A,G1,&D9ED, [12963]
B2,&D9F8,D2,&DA04,F2,&DAE3,C3,&DAE8,D3,&DA
EE,E3,&DBC7,A4,&DBCD,B4,&DBD2,C4,&DBDF,E4,
&DBE4,F4,&DBEA,G4,&DCC3,C5,&DCC8,D5,&DCCE,
E5,&DDAD,B6,&DDB8,D6,&DDC4,F6,&DE97,A7,&DE
A8,D7,&DEBA,G7
285 DATA &D8F7,A1,&DBC7,A4,&DE97,A7,&D9ED, [13712]
B2,&DBCD,B4,&DDAD,B6,&DAE3,C3,&DBD2,C4,&DC
C3,C5,&D908,D1,&D9F8,D2,&DAE8,D3,&DCC8,D5,
&DDB8,D6,&DEA8,D7,&DAEE,E3,&DBDF,E4,&DCCE,
E5,&DA04,F2,&DBE4,F4,&DDC4,F6,&D91A,G1,&DB
EA,G4,&DEBA,G7
286 FOR za=1 TO 16:READ comne$(za):NEXT [2530]
287 DATA A1D1G1,B2D2F2,C3D3E3,A4B4C4,E4F4G [7626]
4,C5D5E5,B6D6F6,A7D7G7,A1A4A7,B2B4B6,C3C4C
5,D1D2D3,D5D6D7,E3E4E5,F2F4F6,G1G4G7
288 FOR za=1 TO 52:READ tp1(za):READ tp2(z [6078]
a):READ tp3(za):READ tp4(za):READ tp5(za):
NEXT
289 DATA 10,2,15,1,3,11,5,14,4,6,12,8,13,7 [12415]
,9,10,23,15,22,24,11,20,14,19,21,12,17,13,
16,18,23,10,2,22,1,20,11,5,19,4,17,12,8,16
,7,2,15,23,3,24,5,14,20,6,21,8,13,17,9,18,
8,1,3,5,2,2,7,9,5,8,12,1,22,11,10,10,7,16,
11,12,17,22,24,20,23,23,16,18,20,17,13,24,
3
290 DATA 14,15,15,18,9,14,13,4,2,7,5,8,4,8 [9006]
,1,5,2,6,2,9,5,8,6,8,3,5,2,6,15,9,14,13,6,
13,3,14,15,21,15,18,14,13,21,13,24,14,15,2
1,23,18,20,17,21,17,24,20,23,19,23,16,20,1
7,19,17,22,20,23,4,10,7,11,12,4,12,1,11,10
,19,10,16,11,12,19,12,22,11,10
291 DATA 10,19,5,11,4,12,19,5,11,4,10,4,20 [9924]
,11,19,12,4,20,11,19,2,4,14,5,6,8,4,14,5,6
,2,6,11,5,4,8,6,11,5,4,15,6,20,14,21,13,6,
20,14,21,15,21,5,14,6,13,21,5,14,6,23,21,1
1,20,19,17,21,11,20,19,23,19,14,20,21,17,1
9,14,20,21
292 FOR za=1 TO 16:READ tp6(za):READ tp7(z [3679]
a):NEXT
293 DATA 6,9,6,3,4,7,4,1,21,18,21,24,6,9,6 [3549]
,3,19,16,19,22,21,18,21,24,19,16,19,22,4,7
,4,1
294 FOR za=1 TO 104:READ tp8(za):READ tp9( [6238]
za):READ tp10(za):READ tp11(za):READ tp12(
za):NEXT
295 DATA 1,5,2,3,8,3,5,2,1,8,1,11,10,22,12 [11399]
,22,11,10,1,12,3,14,15,24,13,24,14,15,3,13
,22,20,23,24,17,24,20,23,22,17,4,8,5,2,6,4
,2,5,8,6,6,8,5,2,4,6,2,5,8,4,4,12,11,10,19
,4,10,11,12,19,19,12,11,10,4,19,10,11,12,4
,6,13,14,15,21,6,15,14,13,21,21,13,14,15,6
296 DATA 21,15,14,13,6,19,23,20,17,21,19,1 [9423]
7,20,23,21,21,23,20,17,19,21,17,20,23,19,1
0,2,1,3,22,10,23,22,1,24,15,2,3,1,24,15,23
,24,3,22,11,5,4,6,19,11,20,19,4,21,14,5,6,
4,21,14,20,21,6,19,12,8,7,9,16,12,17,16,7,
18,13,8,9,7,18,13,17,18,9,16,1,12,10,11,22
297 DATA 22,12,10,11,1,1,8,2,5,3,3,8,2,5,1 [11828]
,3,13,15,14,24,24,13,15,14,3,24,17,23,20,2
2,22,17,23,20,24,1,4,11,10,19,1,19,11,10,4
,22,4,11,10,19,22,19,11,10,4,3,6,14,15,21,
3,21,14,15,6,24,6,14,15,21,24,21,14,15,6,1
,4,5,2,6,1,6,5,2,4,3,4,5,2,6,3,6,5,2,4
298 DATA 22,19,20,21,23,22,21,20,19,23,24, [9734]
19,20,21,23,24,21,20,19,23,1,7,11,10,16,1,
16,11,10,7,1,7,5,2,9,1,9,5,2,7,3,9,14,15,1
8,3,18,14,15,9,3,9,5,2,7,3,7,5,2,9,24,9,14
,15,18,24,18,14,15,9,24,16,20,23,18,24,18,
20,23,16
299 DATA 22,16,11,10,7,22,7,11,10,16,22,16 [10560]
,20,23,18,22,18,20,23,16,1,24,3,2,15,1,24,
22,10,23,4,21,6,5,14,4,21,19,11,20,7,18,9,
8,13,7,18,16,12,17,22,3,1,2,10,22,3,24,15,
23,19,6,4,5,11,19,6,21,14,20,16,9,7,8,12,1
6,9,18,13,17
300 DATA 10,16,11,12,7,10,7,11,12,16,2,7,5 [10225]
,8,9,2,9,5,8,7,15,9,14,13,18,15,18,14,13,9
,23,18,20,17,16,23,16,20,17,18,12,22,11,10
,1,12,1,11,20,22,8,1,5,2,3,8,3,5,2,1,13,3,
14,15,24,13,24,14,15,23,17,24,20,23,22,17,
22,20,23,24
301 FOR za=1 TO 48:READ tp13(za):READ tp14 [4627]
(za):READ tp17(za):NEXT
302 DATA 10,1,0,5,2,0,15,3,0,11,4,0,8,5,2, [7734]
14,6,0,12,7,0,5,8,0,13,9,0,1,10,22,4,11,19
,7,12,16,9,13,18,6,14,21,3,15,24,12,16,0,2
0,17,0,13,18,0,11,19,0,17,20,23,14,21,0,10
,22,0,20,23,0,15,24,0
303 DATA 2,1,0,11,10,0,23,22,0,5,4,0,12,11 [8286]
,10,20,19,0,8,7,0,11,12,0,17,16,0,1,2,3,4,
5,6,7,8,9,16,17,18,19,20,21,22,23,24,8,9,0
,14,13,0,20,18,0,5,6,0,13,14,15,20,21,0,2,
3,0,14,15,0,23,24,0
304 FOR za=1 TO 12:READ tp15(za):READ tp16 [3414]
(za):NEXT
305 DATA 22,24,19,21,16,18,1,3,4,6,7,9,24, [3033]
3,21,6,18,9,1,22,4,19,7,16
306 FOR za=1 TO 3:READ tpsc(za):READ tpwe( [6160]
za):NEXT
307 DATA 735,735,735,495,270,735 [1449]
308 DATA 9,12,16,10,12,15,11,12,14,9,10,11 [5223]
,14,15,16,11,13,14,10,13,15,9,13,16,1,4,8,
2,4,7,3,4,6,1,2,3,6,7,8,3,5,6,2,5,7,1,5,8
309 REM ---------------- UNTERPROGRAMM ´f [5351]
uktionen definieren ----------------
310 DEF FNmu1=PEEK(coalw(1))+PEEK(coalw(2) [12097]
)+PEEK(coalw(3)):DEF FNmu2=PEEK(coalw(4))+
PEEK(coalw(5))+PEEK(coalw(6)):DEF FNmu3=PE
EK(coalw(7))+PEEK(coalw(8))+PEEK(coalw(9))
:DEF FNmu4=PEEK(coalw(10))+PEEK(coalw(11))
+PEEK(coalw(12))
311 DEF FNmu5=PEEK(coalw(13))+PEEK(coalw(1 [3468]
4))+PEEK(coalw(15))
312 DEF FNmu6=PEEK(coalw(16))+PEEK(coalw(1 [12702]
7))+PEEK(coalw(18)):DEF FNmu7=PEEK(coalw(1
9))+PEEK(coalw(20))+PEEK(coalw(21)):DEF FN
mu8=PEEK(coalw(22))+PEEK(coalw(23))+PEEK(c
oalw(24)):DEF FNmu9=PEEK(coalw(1))+PEEK(co
alw(10))+PEEK(coalw(22))
313 DEF FNmu10=PEEK(coalw(4))+PEEK(coalw(1 [11951]
1))+PEEK(coalw(19)):DEF FNmu11=PEEK(coalw(
7))+PEEK(coalw(12))+PEEK(coalw(16)):DEF FN
mu12=PEEK(coalw(2))+PEEK(coalw(5))+PEEK(co
alw(8)):DEF FNmu13=PEEK(coalw(17))+PEEK(co
alw(20))+PEEK(coalw(23))
314 DEF FNmu14=PEEK(coalw(9))+PEEK(coalw(1 [13171]
3))+PEEK(coalw(18)):DEF FNmu15=PEEK(coalw(
6))+PEEK(coalw(14))+PEEK(coalw(21)):DEF FN
mu16=PEEK(coalw(3))+PEEK(coalw(15))+PEEK(c
oalw(24)):POKE &B939,&F3:RETURN
315 frsp=FRE(""):RETURN [1202]
316 mu(1)=FNmu1:mu(2)=FNmu2:mu(3)=FNmu3:mu [14380]
(4)=FNmu4:mu(5)=FNmu5:mu(6)=FNmu6:mu(7)=FN
mu7:mu(8)=FNmu8:mu(9)=FNmu9:mu(10)=FNmu10:
mu(11)=FNmu11:mu(12)=FNmu12:mu(13)=FNmu13:
mu(14)=FNmu14:mu(15)=FNmu15:mu(16)=FNmu16:
RETURN
317 CLOSEIN:PRINT#0," REWIND   TAPE   PLE [3017]
ASE":FOR zei=1 TO 1000:NEXT:CLS#0:RESUME 1
12
318 POKE &B4E8,255:RETURN [1523]
319 muh1=0:FOR za=1 TO 16:IF mu(za)=735 TH [3074]
EN muh1=muh1+1
320 NEXT:RETURN [940]
321 LOCATE#2,2,8:PRINT#2,"GEN. STE.:":LOCA [3728]
TE#3,2,8:PRINT#3,"GEN. STE.:":RETURN
20000 STOP [464]
```



Wipo



Programminhalt:
Bei diesem Programm handelt es sich um eine Simulation des Schiffshandels im 19. Jahrhundert. Dem Privatleben des Schiffsunternehmers und dem geschichtlichen Hintergrund wurde keine Berücksichtigung geschenkt. Dennoch bereitet es viele fröhliche Stunden, wenn man neue Handelspartner findet, regen Warenhandel betreibt oder sich als Schiffsbauer versucht. Ziel des Spieles ist es, alle möglichen Handelspartner zu entdecken und danach den gesamten Welthandel von einem bestimmten Punkt zu steuern.

Programmbeschreibung:
Das Teilprogramm "MC-TEIL" erzeugt drei Maschinenprogramme und die Weltkarte, wie auch das Helpmenue-Fenster. Es empfiehlt sich, das Binärfile direkt hinter dem Hauptprogramm WIPO abzuspeichern. Es gilt folgendes:
CALL &8EFF - Tastatur auf Hauptmenueeingabe abfragen
CALL &8F25 - Weltkarte ein-/ausblenden
CALL &8F2A - Helpmenue ein-/ausblenden
Da das übrige Programm in BASIC geschrieben ist, verweise ich auf die REM-Statements.
Aus programmiertechnischen Gründen ist es nicht möglich, über mehr als 20 Schiffe zu verfügen.

Schiffe senden:
Nach der Schiffswahl erscheint rechts unten die Liste der möglichen Anweisungen, welche nur mit Ziffern ausgewählt werden, links unten kann der Fahrplan, sofern er teilweise oder vielleicht schon ganz erstellt wurde, abgelesen werden. Der Fahrplan wird allerdings nach dessen Ausführung jedesmal gelöscht.

Der Computer erkennt nur das Land an, welches mit "Handelspartner" entdeckt wurde. Um die Eingabe zu erleichtern, wurden folgende Tasten in Verbindung mit CTRL belegt:

D - Deutschland	E - Elfenbeinküste
P - Portugal	I - Indien
U - USA	C - China
M - Mexiko	A - Australien
B - Brasilien	SPACE - Madagaskar

Die Punkte 2 - 8 bedürfen, so hoffe ich, keiner Erklärung, da die Ware mit Cursortasten rechts oben und die Menge eingegeben werden muß. Bei jeder Reparatur des Schiffes muß man eine Zeiteinbuße von einem Monat in Kauf nehmen. Sollten Sie irrsinnige Angaben machen und der Computer hat es in seinem Fahrplan aufgenommen, so wird später bei Bearbeitung auf einen gültigen Wert umgerechnet. Insgesamt sind zwölf Befehle möglich. Beachten Sie bei der Aufstellung, daß das Schiff zum Schluß nicht am Ende der Welt stehen bleibt. Beenden Sie die Eingabe mit "9", so wird nach dem Geld gefragt, das Sie dem Schiff mitgeben. Die Summe spielt dabei keine Rolle, da die Aufträge immer erledigt werden. Es ist also möglich, daß das Schiff mit negativem Kapital zurückkehrt. Mit dieser letzten Eingabe endet die Hauptfunktion.
Hinweis: Wenn Sie zu lange Strecken ohne Pausen und Reparatur befehlen, sinkt Ihr Schiff auf den Meeresgrund.

F = Handelspartner:
Mit dieser Option haben Sie die Möglichkeit, Ihren Aufenthaltsort und den Ort eines Schiffes zu verändern. Am An-

Fortsetzung auf Seite 81.

```
10 REM ------------------------------------ [3706]
------------------------------------
20 REM geschrieben von Richard Krampfl am  [4283]
1.5.86
30 REM Hauptprogramm ´WIPO´ muss nachgelad [4043]
en werden
40 REM ------------------------------------ [3706]
------------------------------------
50 REM ---------------------MC Programm la [3088]
den---------------------------------
60 MEMORY &8EFE:FOR a=&8EFF TO &8F4F:READ  [2409]
i$:POKE a,VAL("&"+i$):NEXT
70 REM -------------------------Weltkarte e [3687]
rstellen----------------------------
80 MODE 2:PLOT 319,399,1:DRAW 638,399:DRAW [8774]
 638,232:DRAW 319,232:DRAW 319,399:PLOT 31
8,399:DRAW 318,232:PLOT 639,232:DRAW 639,3
99:RESTORE 220:x=320:y=397:fa=0:FOR durch=
1 TO 81
90 READ a:IF a>-1 THEN FOR b=1 TO a:PLOT x [6608]
,y,fa:PLOTR 1,0:x=x+2:NEXT:fa=ABS(1-fa):GO
TO 90
100 x=320:y=y-2:fa=0:LOCATE 1,15:NEXT:PLOT [6189]
 473,366,1:DRAWR 0,-4:DRAWR -3,0:DRAWR 0,4
:DRAWR 1,0:DRAWR 0,-4:DRAWR 1,0:DRAWR 0,4:
CALL &8F25
110 REM ---------------------------Hilfsme [4008]
nue---------------------------------
120 MODE 2:PEN 1:PLOT 319,399,1:DRAW 638,3 [13503]
99:DRAW 638,232:DRAW 319,232:DRAW 319,399:
PLOT 318,399:DRAW 318,232:PLOT 639,232:DRA
W 639,399:LOCATE 1,25:PRINT"Hilfsmenue":FO
R a=0 TO 80:FOR b=0 TO 14 STEP 2:PLOT 402+
a*2,b+374,TEST(a,b):PLOTR 1,0:NEXT b,a
130 FOR a=0 TO 4:READ i$:LOCATE 1,25:PRINT [6122]
 i$:FOR b=0 TO 6:FOR c=1 TO 15 STEP 2:PLOT
 b*2+323,352-a*18+c,TEST(b,c):PLOTR 1,0
140 NEXT c,b:READ i$:TAG:PLOT 700,0,1:MOVE [11624]
 337,366-a*18:PRINT " = ";i$;:TAGOFF:READ
i$:LOCATE 1,25:PRINT i$:FOR b=0 TO 6:FOR c
=1 TO 15 STEP 2:PLOT b*2+484,352-a*18+c,TE
ST(b,c):PLOTR 1,0:NEXT c,b:READ i$:TAG:PLO
T 700,0,1:MOVE 498,366-a*18
150 IF a<4 THEN PRINT " = ";i$;:TAGOFF:NEX [1622]
T
160 TAG:MOVE 323,274:PRINT"(Diese Tasten s [10986]
tehen nur in Verbindung";:TAGOFF:LOCATE 42
,10:PRINT"mit Shift zur Verfuegung.)":CALL
 &8F2A:MODE 2:SPEED WRITE 1
170 PRINT"Der MC-Teil wird binaer abgespei [11511]
chert.":SAVE "fenster",b,&8EFF,&1C81:PRINT
"Bei weiterem Abspeichern nur ´goto 170´ e
ingeben.":END
180 REM -------------------------MC Progra [3402]
mm----------------------------------
190 DATA 06,09,11,1B,8f,1A,CD,1E,BB,20,04, [5917]
13,10,F7,C9,3E,15,CD,1E,BB,28,F5,78,32,24,
8f,18,EF,2c,47,35,1b,3c,24,33,37,3a,00
200 DATA 11,68,9d,18,03,11,50,8f,21,27,c0, [6268]
0e,0b,c5,06,29,e5,7e,eb,4e,eb,71,12,23,13,
10,f6,e1,01,00,08,09,30,ec,01,50,c0,09,c1,
0d,20,e3,c9
210 REM -------------------------DATA´s-- [3621]
------------------------------------
220 DATA 11,1,4,4,3,1,11,2,5,1,1,1,9,1,12, [3039]
1,29,2,2,2,6,32,2,1,15,-1
230 DATA 8,1,2,1,4,1,7,7,12,1,1,2,7,1,11,1 [2941]
,30,1,1,2,40,1,2,2,13,-1
240 DATA 0,1,1,1,2,3,1,2,3,2,15,1,8,1,2,2, [3194]
2,1,6,1,9,2,31,1,42,1,5,1,5,7,-1
250 DATA 0,2,1,2,7,2,18,2,1,2,2,1,1,1,3,1, [3290]
2,1,5,1,8,3,16,2,10,1,2,1,43,1,5,2,4,1,5,-
1
260 DATA 9,3,22,1,2,2,2,1,3,1,2,1,5,1,5,3, [2365]
17,7,7,4,38,1,3,1,8,5,5,-1
270 DATA 9,2,28,3,2,1,4,1,3,2,3,3,3,5,9,2, [2778]
6,2,1,6,40,1,1,1,1,1,13,2,4,-1
280 DATA 10,1,2,5,22,1,3,2,1,3,3,1,3,1,6,2 [4596]
,2,1,8,1,3,1,4,4,41,5,1,1,1,1,14,1,4,-1

290 DATA 7,3,1,2,5,1,17,4,6,1,6,2,1,1,9,2, [4107]
8,1,3,1,1,1,4,3,41,1,3,1,1,1,2,1,13,1,4,-1

300 DATA 6,5,7,1,16,1,7,3,8,2,19,2,2,1,1,1 [2880]
,49,1,5,1,2,1,1,1,16,-1
310 DATA 5,1,12,1,16,1,5,2,2,3,26,1,4,1,1, [2579]
2,47,1,7,2,5,1,5,1,1,1,5,-1
320 DATA 18,1,16,2,3,1,6,1,22,2,2,1,1,1,2, [3539]
1,3,2,46,3,1,1,3,2,7,1,10,-1
```

```
330 DATA 18,1,18,1,2,1,6,1,22,2,3,2,2,1,2, [2057]
1,51,1,1,1,3,1,18,-1
340 DATA 17,1,19,1,2,1,6,1,20,1,2,2,4,2,2, [1776]
1,52,1,1,2,21,-1
350 DATA 17,1,19,1,1,1,7,2,18,2,1,3,2,1,3, [3210]
3,53,1,1,2,20,-1
360 DATA 17,1,20,1,9,1,18,2,1,3,1,5,56,1,2 [1989]
,2,19,-1
370 DATA 16,2,15,1,9,5,25,1,62,1,22,-1      [1362]

380 DATA 16,1,14,4,7,2,27,2,63,1,2,2,18,-1 [1767]

390 DATA 15,1,16,3,7,1,1,2,26,1,14,1,2,1,5 [1839]
,2,36,3,2,2,18,-1
400 DATA 14,1,17,1,1,2,4,5,27,1,5,3,5,1,1, [3458]
3,4,1,1,1,36,1,4,1,19,-1
410 DATA 14,1,17,1,2,2,2,1,28,5,1,6,1,1,5, [3728]
1,3,1,3,1,1,1,33,1,2,1,4,2,18,-1
420 DATA 13,1,24,1,29,1,4,1,4,3,1,1,1,2,2, [3220]
4,4,1,1,1,31,5,4,2,18,-1
430 DATA 13,1,23,1,30,1,3,1,6,3,1,2,1,1,9, [3208]
1,1,1,31,1,3,2,2,1,1,1,18,-1
440 DATA 12,1,23,1,20,1,1,1,8,1,2,1,8,1,2, [2527]
1,1,1,10,1,1,1,32,1,2,2,2,1,1,1,18,-1

450 DATA 12,1,22,1,33,3,1,4,2,1,3,1,2,4,7, [2055]
2,32,2,2,1,1,1,2,1,18,-1
460 DATA 12,1,20,2,34,4,4,1,12,1,42,1,3,1, [2224]
1,2,18,-1
470 DATA 13,1,18,1,30,1,4,1,8,2,11,1,42,1, [1893]
4,1,20,-1
480 DATA 14,2,9,2,1,4,35,1,11,1,3,2,5,1,42 [2631]
,1,25,-1
490 DATA 14,2,8,1,2,2,1,2,2,1,27,1,4,1,12, [2601]
3,2,5,7,1,36,1,24,-1
500 DATA 14,3,7,1,5,2,31,1,2,1,29,1,1,1,35 [1692]
,1,24,-1
510 DATA 14,2,1,1,5,1,6,2,29,1,3,1,23,2,6, [2916]
1,1,3,32,1,24,-1
520 DATA 15,1,1,1,5,1,11,1,28,1,24,1,1,1,6 [2686]
,1,3,5,26,1,2,1,22,-1
530 DATA 15,1,1,1,5,1,5,1,33,1,26,1,1,1,5, [2718]
4,5,2,8,1,2,1,11,1,3,1,22,-1
540 DATA 17,1,5,1,2,2,2,3,30,1,26,1,1,1,9, [3096]
2,4,3,6,3,1,1,6,4,27,-1
550 DATA 17,1,5,3,1,1,4,2,29,1,27,1,1,1,9, [3181]
1,6,1,5,1,4,1,6,1,30,-1
560 DATA 17,2,8,1,6,2,2,1,24,1,27,1,2,1,7, [2222]
1,7,1,5,1,4,1,6,1,30,-1
570 DATA 19,1,8,1,28,1,5,1,27,1,2,1,6,1,9, [3237]
1,3,1,6,2,5,1,6,1,22,-1
580 DATA 20,2,7,1,11,1,21,1,28,1,2,1,4,1,1 [3821]
0,1,2,1,9,1,5,1,4,2,22,-1
590 DATA 21,3,5,1,29,1,2,1,30,1,1,5,11,1,2 [3111]
,1,9,3,3,1,5,2,21,-1
600 DATA 24,3,2,1,33,1,30,1,17,1,2,1,8,3,3 [2151]
,1,5,1,1,1,20,-1
610 DATA 26,1,2,1,3,3,6,1,20,1,31,5,12,1,1 [2400]
,1,9,2,1,1,1,1,7,1,1,1,19,-1
620 DATA 26,1,3,1,1,1,2,5,24,1,35,1,9,1,1, [3015]
1,1,1,9,2,2,1,5,1,2,1,1,1,19,-1
630 DATA 27,3,1,1,8,1,24,1,34,1,12,1,1,1,8 [2718]
,2,12,1,1,1,15,1,2,-1
640 DATA 29,2,10,1,23,2,5,3,25,1,14,1,10,1 [1586]
,11,3,7,1,10,-1
650 DATA 30,1,11,2,23,5,3,1,24,1,11,1,14,1 [2465]
,16,1,14,-1
660 DATA 30,1,13,2,30,1,22,1,27,2,4,3,16,1 [2009]
,6,-1
670 DATA 30,1,14,1,28,1,1,1,22,1,12,1,10,2 [2218]
,8,1,1,1,23,-1
680 DATA 30,1,14,1,30,1,12,1,9,1,24,2,6,1, [2245]
2,1,6,1,16,-1
690 DATA 24,1,4,1,15,1,13,1,14,1,1,1,11,2, [3469]
8,1,26,2,4,1,3,1,2,1,20,-1
700 DATA 29,1,16,1,30,1,19,1,28,2,3,1,3,1, [1852]
23,-1
710 DATA 29,1,17,5,26,1,17,1,30,2,2,1,3,1, [2109]
7,2,8,1,5,-1
720 DATA 29,1,22,3,23,1,17,1,30,2,3,3,4,1, [2014]
3,5,11,-1
730 DATA 29,1,25,1,22,1,16,1,31,2,16,1,2,1 [1881]
,10,-1
740 DATA 29,1,25,1,22,1,16,1,16,1,17,1,15, [1352]
1,2,2,3,1,4,-1
750 DATA 30,1,24,1,23,1,14,1,8,1,27,6,4,1, [2341]
4,1,3,1,8,-1
760 DATA 31,1,22,1,25,1,13,1,38,4,9,6,7,-1 [1857]
770 DATA 31,1,21,1,26,1,13,1,7,1,35,1,11,2 [2300]
,7,-1
780 DATA 32,1,20,1,27,1,13,1,45,1,1,1,2,2, [3014]
6,1,4,-1
790 DATA 33,1,19,1,27,1,13,1,6,1,37,1,1,2, [2132]
2,2,11,-1
800 DATA 34,1,18,1,26,1,14,1,2,1,38,4,2,1, [2091]
2,2,11,-1
810 DATA 35,1,17,1,26,1,14,1,5,1,34,1,7,2, [2578]
2,1,3,1,6,-1
820 DATA 35,1,16,1,27,1,14,1,3,2,1,1,33,1, [2335]
7,2,2,1,10,-1
830 DATA 35,1,16,1,27,1,13,1,4,1,1,1,32,2, [2298]
12,1,10,-1
840 DATA 36,1,13,3,27,1,12,1,4,1,1,1,4,1,2 [2841]
5,3,15,1,6,1,2,-1
850 DATA 36,1,12,1,31,1,10,1,5,1,1,1,29,1, [2511]
19,1,8,-1
860 DATA 36,1,12,1,31,1,10,1,5,2,6,1,23,1, [2318]
19,1,8,-1
870 DATA 36,1,11,1,32,1,10,1,5,2,30,1,19,1 [1744]
,8,-1
880 DATA 36,1,11,1,32,1,10,1,5,2,30,1,19,1 [1744]
,8,-1
890 DATA 36,1,11,1,33,1,8,1,38,1,19,1,8,-1 [1802]
900 DATA 36,1,10,1,34,1,8,1,38,1,19,1,8,-1 [2207]
910 DATA 36,1,10,1,35,1,5,2,39,1,19,1,8,-1 [1656]
920 DATA 36,1,8,2,36,1,4,1,41,1,5,4,9,1,9, [1826]
-1
930 DATA 36,1,8,1,38,4,43,5,4,1,7,1,10,-1  [1606]
940 DATA 36,1,7,2,94,1,6,1,11,-1            [1664]
950 DATA 36,1,7,1,95,1,4,2,9,2,1,-1         [1117]
960 DATA 37,2,3,2,97,4,11,2,1,-1            [1096]
970 DATA 38,1,4,1,111,1,2,1,-1              [1366]
980 DATA 38,1,3,2,98,2,10,1,1,2,1,-1        [1080]
990 DATA 38,1,4,1,98,2,9,1,1,1,3,-1         [1226]
1000 DATA 39,2,2,1,3,1,103,4,4,-1           [1204]
1010 DATA 40,3,4,2,101,1,1,1,6,-1           [881]
1020 DATA 41,3,105,3,7,-1                   [1150]
1030 REM --------------------------------- [3732]
Datas-------------------------------
1040 DATA E,Wareneinkauf,P,Schiffe planen, [8611]
V,Warenverkauf,S,Schiffe senden,T,Warentra
nsfer,F,Handelspartner,L,Lager bauen,Z,Zuk
unft,H,Hilfsmenue," ",
```

```
10 REM --------------------------------     [2043]
20 REM                   WIPO               [835]
30 REM                                      [272]
40 REM written by Richard Krampfl           [2061]
50 REM at May / June 1986                   [767]
60 REM --------------------------------     [2043]
70 '                                        [117]
80 ' Dimensionierungen                      [2081]
90 '                                        [117]
100 IF HIMEM<>36606 THEN MEMORY &8EFE:PRIN [7147]
T:PRINT"Es wird 'Fenster' nachgeladen !":P
RINT:LOAD "Fenster":CLEAR
110 DEFSTR m-t:DEFINT b:DIM q(12),m(10),j( [7494]
20,10),jb(10,10),fi(20),l(10),b(10,10),n(1
0,7),s(20),k(20),i(20),e(10,10),a(20),d(20
,13),t2(10),f(10)
120 GOSUB 1290:g=0:EVERY 5,3 GOSUB 160:EVE [6179]
RY 149,2 GOSUB 200:WHILE g=0:WEND:CLS#2:CL
S#3:GOTO 180
130 '                                       [117]
140 ' Hauptmenue-Abfrage                    [1791]
150 '                                       [117]
160 IF g=0 THEN POKE &8F24,0:CALL &8EFF:g= [2446]
PEEK(&8F24)
170 RETURN                                  [555]
180 ON g GOTO 2240,2390,2030,2490,990,2590 [2397]
,1750,1420,240:GOTO 180
190 REM Carbage Collection                  [1418]
200 RETURN:cc=FRE(" "):RETURN               [1873]
210 '                                       [117]
220 REM Hauptmenue einblenden               [2271]
230 '                                       [117]
240 IF f=1 THEN CALL &8F25:CALL &8F2A ELSE [3691]
 IF f=0 THEN CALL &8F2A
250 f=2:g=0:WHILE g=0:WEND:IF g=10 THEN 25 [3804]
0 ELSE CALL &8F2A:f=0:GOTO 180
260 REM Weltkarte ausblenden                [1892]
270 IF f=1 THEN CALL &8F25:f=0              [1749]
280 RETURN                                  [555]
290 REM Datumsanzeige                       [656]
300 IF (z-INT(z))*100>12.1 THEN z=z+0.88:G [2900]
OTO 300
310 LOCATE 10,1:PRINT q((z-INT(z))*100);IN [1756]
```

```
T(z)"        ":RETURN
320 REM Ortsanzeige                                      [1432]
330 LOCATE 10,2:PRINT SPACE$(16):LOCATE 10 [2781]
,2:PRINT l$(1):RETURN
340 REM Geldanzeige                                      [1849]
350 LOCATE 10,3:PRINT USING"#######";e:RET [2719]
URN
360 REM freier Schiffsraum                               [1147]
370 ab=0:FOR a=1 TO h:IF l=fi(a) THEN ab=a [2946]
b+k(a):e=e+j(a,0):j(a,0)=0:GOSUB 350
380 NEXT a:LOCATE 20,4:PRINT USING"#### T" [2331]
;ab:RETURN
390 REM freier Lagerraum                                 [689]
400 LOCATE 19,5:IF l(1)<0 THEN PRINT"  --- [3941]
--":RETURN ELSE PRINT USING "##### T";l(1)
:RETURN
410 REM Schiffe im Hafen                                 [1485]
420 LOCATE 40,1:PRINT"Schiffe im Hafen:":L [9303]
OCATE 40,2:FOR a=1 TO h:IF l=fi(a) THEN IF
 POS(#0)+2+LEN(s(a))>80 THEN LOCATE 40,VPO
S(#0)+1:PRINT s(a)", "; ELSE PRINT s(a)",
";
430 NEXT a:PRINT CHR$(8)CHR$(8)" ";:RETURN [1293]
440 REM Eingabe von Ware und Menge                       [1384]
450 GOSUB 270:CLS#1:FOR a3=1 TO 10:LOCATE [3587]
50,a3:PRINT m(a3):NEXT:a3=1
460 LOCATE 50,a3:PRINT CHR$(24)+m(a3)+CHR$ [2412]
(24)
470 t=INKEY$:IF t=CHR$(240) AND a3>1 THEN  [9761]
GOSUB 510:a3=a3-1:GOTO 460 ELSE IF t=CHR$(
241) AND a3<10 THEN GOSUB 510:a3=a3+1:GOTO
 460 ELSE IF t=CHR$(224) THEN a1=a3:GOTO 4
80 ELSE 470
480 INPUT#3,"Menge";a2:f=1:CALL &8F25:RETU [3119]
RN
490 REM Taetigkeiten von Land a(0) verschi [2497]
eben
500 FOR a5=1 TO 6:n(a(0),a5)=n(a(0),a5+1): [4724]
NEXT:n(a(0),7)=DEC$((z,"####.##")+":  ":RET
URN
510 LOCATE 50,a3:PRINT m(a3):RETURN                      [2124]
520 REM Waren eines Schiffes printen                     [903]
530 LOCATE 1,12:PRINT"Die Waren des Schiff [8727]
es "s(a)":":FOR a1=1 TO 10:IF j(a,a1)>0 AN
D POS(#0)+LEN(m(a1))+LEN(STR$(j(a,a1)))+4>
80 THEN LOCATE 1,VPOS(#0)+1
540 IF j(a,a1)>0 THEN PRINT j(a,a1)"T "m(a [3226]
1)" "CHR$(149);
550 NEXT a1:PRINT CHR$(8)" ":RETURN                      [2321]
560 REM Monatsablauf in Prozent                          [1979]
570 DI:IF a(0)>=100 THEN z=z+0.01:GOSUB 30 [10205]
0:a(0)=a(0)-100:a(0)=INT(a(0)/4)*4:GOSUB 1
440:LOCATE 41,17:PRINT SPACE$(30):PLOT 320
,126,1:DRAWR 0,18:DRAWR 202,0:DRAWR 0,-18:
GOTO 570
580 PLOT 320+a(0)*2,126,1:DRAWR 0,18:DRAWR [5423]
 2,0:DRAWR 0,-18:DRAWR 2,0:DRAWR 0,18:DRAW
R 2,0:DRAWR 0,-18:i(a(1))=i(a(1))+0.5
590 REM Abnuetzung des Schiffes in %                     [2324]
600 DI:IF i(a(1))>66 THEN t="Sie ertranken [10563]
, da das Schiff nicht mehr seetuechtig ist
.":GOTO 2180 ELSE PLOT 320+i(a(1))*2,94,1:
DRAWR 0,18:RETURN
610 REM JA/Nein Abfrage                                  [798]
620 WHILE INKEY$<>"":WEND                                [1786]
630 t=LOWER$(INKEY$):IF t<>"j" AND t<>"n"  [1887]
THEN 630 ELSE RETURN
640 REM Eingabetext printen                              [2090]
650 PRINT#3,"-Warte auf Eingabe-":RETURN                 [2092]
660 REM Blinken simulieren                               [1166]
670 PLOT x,y,ABS(1-TEST(x,y)):PLOTR 1,0:RE [2904]
TURN
680 REM Taetigkeiten von Land verschieben                [1919]
690 FOR a4=1 TO 6:n(1,a4)=n(1,a4+1):NEXT:n [3764]
(1,7)=DEC$((z,"####.##")+":  ":RETURN
700 REM Schiff auswaehlen                                [2348]
710 CLS#4:FOR a=1 TO 10 STEP 2:LOCATE a*7+ [10107]
8-LEN(m(a))/2,12:PRINT m(a):LOCATE a*7+15-
LEN(m(a+1))/2,13:PRINT m(a+1):NEXT:LOCATE
72,13:PRINT m(10);:PRINT STRING$(80,154):a
(0)=0:a=0:LOCATE 1,15
720 WHILE VPOS(#0)<25 AND a<h:a=a+1:IF fi( [9661]
a)=1 THEN PRINT s(a);:LOCATE 11,VPOS(#0):F
OR a1=1 TO 10:PRINT USING"    ####";j(a,a1)
;:NEXT:a(0)=a(0)+1:a(a(0))=a
730 WEND:a=1:g=0                                         [581]
740 LOCATE 1,14+a:PRINT CHR$(24)+s(a(a))+C [3348]
HR$(24)
750 IF g>0 THEN RETURN ELSE t=INKEY$:IF t< [10570]
CHR$(224) THEN 750 ELSE IF ASC(t)=240 AND
a>1 THEN GOSUB 760:a=a-1:GOTO 740 ELSE IF
ASC(t)=241 AND a<a(0) THEN GOSUB 760:a=a+1
:GOTO 740 ELSE IF ASC(t)=224 THEN a=a(a):C
LS#4:RETURN ELSE 750
760 LOCATE 1,14+a:PRINT s(a(a)):RETURN                   [2846]
770 REM Ware auswaehlen                                  [978]
780 FOR a1=1 TO 10:LOCATE 1,15+a1:PRINT US [12992]
ING"\         \  #####          #####";m(a1
);j(a,a1);jb(1,a1):NEXT:a1=1:LOCATE 1,13:P
RINT"ausgewaehltes Schiff: ";s(a):LOCATE 1
5,15:PRINT"Schiff        Lager"
790 LOCATE 1,a1+15:PRINT USING CHR$(24)+"\ [6424]
        \  #####           #####"+CHR$(24);
m(a1);j(a,a1);jb(1,a1)
800 IF g>0 THEN RETURN ELSE t=INKEY$:IF t< [11406]
CHR$(224) THEN 800 ELSE IF ASC(t)=240 AND
a1>1 THEN GOSUB 810:a1=a1-1:GOTO 790 ELSE
IF ASC(t)=241 AND a1<10 THEN GOSUB 810:a1=
a1+1:GOTO 790 ELSE IF ASC(t)=224 THEN RETU
RN ELSE 800
810 LOCATE 1,a1+15:PRINT USING"\          \ [5817]
  #####           #####";m(a1);j(a,a1);jb(1,
a1):RETURN
820 REM Taetigkeiten printen                             [2288]
830 GOSUB 270:CLS#1:LOCATE 40,1:PRINT"Taet [11387]
igkeiten von Land:":FOR a3=1 TO 7:IF n(1,a
3)<>"" THEN LOCATE 40,VPOS(#0)+1:a2=VAL(LE
FT$(n(1,a3),7)):PRINT USING"##.####";(a2-I
NT(a2))*100+a2/10000;:PRINT RIGHT$(n(1,a3)
,LEN(n(1,a3))-7);
840 NEXT:RETURN                                          [940]
850 REM Freier Schiffsraum von jedem Schif [3126]
f printen
860 GOSUB 270:CLS#1:LOCATE 40,1:PRINT"Frei [15180]
er Schiffsraum:";:a3=0:WHILE VPOS(#0)<12 A
ND a3<h:a3=a3+1:IF fi(a3)=1 THEN LOCATE 40
,VPOS(#0)+1:PRINT USING"\           \   #
#####";s(a3);k(a3);
870 WEND:RETURN                                          [935]
880 REM auf Hauptmenueeingabe warten                     [2120]
890 g=0:GOSUB 830:GOSUB 930:GOSUB 650:WHIL [4795]
E g=0:WEND:GOTO 180
900 REM Preistabelle eines Landes                        [2155]
910 GOSUB 270:CLS#1:LOCATE 40,1:PRINT"Folg [12272]
ende Preise gelten in diesem Land:":FOR a1
=1 TO 10:LOCATE 42,a1+1:PRINT m(a1)STRING$
(15-LEN(m(a1)),46);:PRINT USING"###";b(1,a
1):NEXT:RETURN
920 REM Preistabelle aller Laender                       [1903]
930 CLS#4:FOR a=1 TO 10 STEP 2:LOCATE a*6+ [13739]
18-LEN(m(a))/2,13:PRINT m(a):LOCATE a*6+25
-LEN(m(a+1))/2,14:PRINT m(a+1):NEXT:LOCATE
 72,14:PRINT m(10);:PRINT STRING$(80,154):
FOR a=1 TO 10:LOCATE 1,a+15:IF l(a)<0 THEN
 PRINT STRING$(15,45):GOTO 950
940 PRINT l$(a);:FOR a1=1 TO 10:LOCATE 17+ [4511]
a1*6,VPOS(#0):PRINT USING"###";b(a,a1);:NE
XT
950 IF g>0 THEN RETURN ELSE NEXT:RETURN                  [1969]
960 '                                                    [117]
970 ' Schiffe senden                                     [1438]
980 '                                                    [117]
990 IF l(1)<0 THEN PRINT#3,"Von hier aus n [5755]
icht moeglich.":GOTO 890 ELSE GOSUB 270:CA
LL &8F25:f=1
1000 ab=0:FOR a=1 TO h:ab=ab-(fi(a)=1):NEX [8663]
T:IF ab=0 THEN PRINT#3,"Kein Schiff im Haf
en":GOTO 890 ELSE PRINT#3,"Waehlen Sie das
 Schiff.":GOSUB 710:ga=g:g=0
1010 CLS#4:CLS#7:PRINT#7,"Moegliche Anweis [11605]
ungen:":PRINT#7:PRINT#7,"(1)  Fahre nach
..":PRINT#7,"(2)  Kaufe ...":PRINT#7,"(3)
 Verkaufe ...":PRINT#7,"(4)  Bringe ... vo
m Lager aufs Schiff"
1020 PRINT#7,"(5)  Bringe ... vom Schiff i [8318]
ns Lager":PRINT#7,"(6)  Repariere Schiff":
PRINT#7,"(7)  Warte auf Schiff ...":PRINT#
7,"(8)  Letzteres loeschen"
1030 PRINT#7,"(9)  Eingaben beenden":PRINT [13618]
#7:PRINT#7,"Geben Sie dabei nur die Zahl e
in !":LOCATE 1,12:PRINT CHR$(24)+" Der Fah
rplan des Schiffes "+s(a)+":  "+CHR$(24):a(
1)=a:a=1:WHILE d(a(1),a)>0:a=a+1:WEND:a=a-
1
1040 FOR a3=1 TO a:LOCATE 1,13+a3:PRINT ST [10022]
R$(a3)+"......";:a2=d(a(1),a3)-INT(d(a(1),
a3)/100)*100:ON INT(d(a(1),a3)/100)-1 GOTO
 1050,1060,1070,1080,1090,1100:PRINT"Fahre
```

LISTINGS

```
 nach "+l$(a2):GOTO 1110
1050 PRINT"Kaufe"+STR$(ROUND((a2-INT(a2))* [5413]
10000))+" T "+m(INT(a2)):GOTO 1110
1060 PRINT"Verkaufe"+STR$(ROUND((a2-INT(a2 [4937]
))*10000))+" T "+m(INT(a2)):GOTO 1110
1070 PRINT"Bringe"+STR$(ROUND((a2-INT(a2)) [6949]
*10000))+" T "+m(INT(a2))" aufs Schiff":GO
TO 1110
1080 PRINT"Bringe"+STR$(ROUND((a2-INT(a2)) [6474]
*10000))+" T "+m(INT(a2))" ins Lager":GOTO
 1110
1090 PRINT"Repariere Schiff":GOTO 1110      [2752]
1100 PRINT"Warte auf Schiff "+s(a2)        [2573]
1110 NEXT                                  [350]
1120 GOSUB 650                             [883]
1130 IF g>0 THEN 180 ELSE t=INKEY$:IF t="" [10628]
 THEN 1130 ELSE IF ASC(t)<49 OR ASC(t)>57
THEN 1130 ELSE IF a>11 AND ASC(t)<56 THEN
1130 ELSE a2=ASC(t)-48:ON a2 GOTO 1140,116
0,1170,1180,1190,1200,1210,1230,1250
1140 INPUT#3,"Land";t:a1=0:FOR a3=1 TO 10: [5180]
IF LOWER$(t)=LOWER$(l$(a3)) AND l(a3)>-1 T
HEN a1=a3:a3=10
1150 NEXT:IF a1=0 THEN PRINT#3,"Kenne ich  [5927]
nicht !":GOTO 1120 ELSE a=a+1:d(a(1),a)=10
0+a1:GOTO 1040
1160 GOSUB 450:IF a2=0 THEN 1120 ELSE a=a+ [4856]
1:d(a(1),a)=200+a1+a2/10000:GOTO 1040
1170 GOSUB 450:IF a2=0 THEN 1120 ELSE a=a+ [5570]
1:d(a(1),a)=300+a1+a2/10000:GOTO 1040
1180 PRINT#3,"Geben Sie den Artikel ein.": [6881]
GOSUB 450:IF a2=0 THEN 1120 ELSE a=a+1:d(a
(1),a)=400+a3+a2/10000:GOTO 1040
1190 PRINT#3,"Geben Sie den Artikel ein.": [7193]
GOSUB 450:IF a2=0 THEN 1120 ELSE a=a+1:d(a
(1),a)=500+a3+a2/10000:GOTO 1040
1200 a=a+1:d(a(1),a)=600:GOTO 1040         [1100]
1210 INPUT#3,"Schiff";t:a1=0:FOR a3=1 TO h [4350]
:IF LOWER$(t)=LOWER$(s(a3)) THEN a1=a3:a3=
h
1220 NEXT:IF a1=0 THEN PRINT#3,"Dieses Sch [6341]
iff existiert nicht !":GOTO 1120 ELSE a=a+
1:d(a(1),a)=700+a1:GOTO 1040
1230 IF a>0 THEN LOCATE 1,13+a:PRINT SPACE [4068]
$(39);:d(a(1),a)=0:a=a-1
1240 GOTO 1040                             [363]
1250 PRINT#3,"Wieviel Geld geben Sie dem S [9960]
chiff mit";:INPUT#3,"Summe";a1:j(a(1),0)=j
(a(1),0)+a1:e=e-a1:GOSUB 350:GOTO 890
1260 '                                     [117]
1270 ' Titelbild                           [797]
1280 '                                     [117]
1290 MODE 2:INK 0,1:INK 1,24:BORDER 1:WIND [12023]
OW#1,40,80,1,11:WINDOW#2,1,80,12,23:WINDOW
 #3,2,38,7,10:WINDOW#4,1,80,12,25:WINDOW#5
,41,80,14,19:WINDOW#6,1,80,21,25:WINDOW#7,
42,80,12,25
1300 RESTORE 2890:FOR a=1 TO 12:READ q(a): [10619]
NEXT:FOR a=1 TO 10:READ m(a),l$(a):NEXT:RE
AD l$(0):FOR a=1 TO 10:FOR a1=1 TO 10:READ
 b(a,a1):NEXT a1,a:FOR a=1 TO 9:FOR a1=a T
O 9:READ e(a,a1):NEXT a1,a
1310 FOR a=0 TO 10:l(a)=-1:NEXT:FOR a=1 TO [13792]
 10:KEY 127+a,l$(a):READ a1,a2:KEY DEF a1,
1,ASC(LOWER$(LEFT$(l$(a),1))),ASC(UPPER$(L
EFT$(l$(a),1))),a+127:KEY DEF a2,1,47+a:NE
XT:KEY DEF 47,0,32,32,131:' CTRL + Leertas
te = Madagaskar
1320 x=472:y=370:l=1:e=4500:s(1)="Ostkutte [9003]
r":k(1)=7:i(1)=70:fi(1)=1:l(1)=2:jb(1,3)=4
:jb(1,4)=1:jb(1,5)=6:z=1837.06:h=1:i(1)=27
:l(0)=-1:jl=ROUND(1870+RND*40,2)
1330 PRINT"Datum:":GOSUB 300:PRINT"Ort:":G [10175]
OSUB 330:PRINT"Bargeld:":GOSUB 350:PRINT"f
reier Schiffsraum:":GOSUB 370:PRINT"freier
 Lagerraum:":GOSUB 400:GOSUB 420
1340 LOCATE 1,6:PRINT STRING$(39,154):LOCA [10728]
TE 1,11:PRINT STRING$(39,154):FOR a=1 TO 4
:LOCATE 1,6+a:PRINT CHR$(138):LOCATE 39,6+
a:PRINT CHR$(133):NEXT:LOCATE 1,6:PRINT CH
R$(136):LOCATE 39,6:PRINT CHR$(132)
1350 LOCATE 1,11:PRINT CHR$(130):LOCATE 39 [3187]
,11:PRINT CHR$(129)
1360 LOCATE 10,8:PRINT"Beginnen Sie bitte  [17674]
!!":LOCATE 10,10:PRINT"SHIFT-H  =  Hilfsme
nue":LOCATE 1,13:PRINT"Sie wurden am 17.04
.1809 in Norddeutschland geboren.":PRINT"I
hr Vater betreibt ein Fischunternehmen mit
 maessigem Erfolg."
1370 PRINT"Leider ist Ihr Vater vor einem  [22145]
Monat im Alter von 59 Jahren verstorben.":
PRINT"Sie uebernehmen nun das Geschaeft vo
n Ihrem Vater und bilden sich in Ihrer":PR
INT"Kindheit ein, dass Sie als Schiffsbaue
r mehr Chancen haben."
1380 PRINT"Leider fehlt Ihnen das Anfangsk [4378]
apital !":RETURN
1390 '                                     [117]
1400 ' Zukunft                             [1106]
1410 '                                     [117]
1420 PRINT#3," Beachten Sie die Entwicklun [18370]
g !":IF e<-100 THEN a3=INT(e/-100):PRINT#3
,"Sie muessen"+STR$(a3)+" Monate aussetzen
,":PRINT#3,"da Sie Schulden haben.":PRINT#
3,"Bitte warten":FOR ac=1 TO a3:GOSUB 1440
:z=z+0.01:GOSUB 300:NEXT
1430 g=0:z=z+0.01:GOSUB 300:GOSUB 1440:IF  [3025]
g>0 THEN 180 ELSE 890
1440 FOR a1=1 TO 10:IF l(a1)<-1 THEN l(a1) [2288]
=l(a1)+1
1450 IF ll(a1)>1 THEN ll(a1)=ll(a1)-1:IF l [4634]
l(a1)<1 THEN l(a1)=l(a1)+ROUND(ll(a1)*1000
0):ll(a1)=0:GOSUB 400
1460 IF f(a1)>1 THEN f(a1)=f(a1)-1:IF f(a1 [10128]
)<1 THEN h=h+1:fi(h)=a1:k(h)=ROUND(f(a1)*1
0000):s(h)=t2(a1):f(a1)=-1:FOR a2=0 TO 10:
j(h,a2)=0:NEXT:GOSUB 370
1470 NEXT a1:FOR a1=1 TO h                 [1999]
1480 IF fi(a1)<0 THEN fi(a1)=fi(a1)+1:i(a1 [4647]
)=i(a1)+10:IF fi(a1)>-1 THEN fi(a1)=ROUND(
fi(a1)*-100)
1490 IF i(a1)>66 THEN PRINT#3,"Das Schiff  [9262]
"+s(a1)+" ist gesunken,":PRINT#3,"da es ni
cht mehr seetuechtig war."
1500 IF i(a1)>66 THEN FOR a3=a1 TO h-1:fi( [10451]
a3)=fi(a3+1):FOR a2=1 TO 10:j(a3,a2)=j(a3+
1,a2):NEXT:i(a3)=i(a3+1):k(a3)=k(a3+1):s(a
3)=s(a3+1):NEXT:h=h-1:GOSUB 350
1510 a2=0:IF fi(a1)<0 THEN 1690            [1651]
1520 IF d(a1,1)=-1 THEN 1690               [762]
1530 WHILE d(a1,a2)<1 AND a2<12:a2=a2+1:WE [11386]
ND:a3=d(a1,a2)-INT(d(a1,a2)/100)*100:a4=RO
UND((a3-INT(a3))*10000):a3=INT(a3):a(0)=fi
(a1):ON INT(d(a1,a2)/100) GOTO 1540,1550,1
580,1630,1660,1610,1620:d(a1,1)=-1:GOTO 16
90
1540 fi(a1)=-e(MIN(fi(a1),a3),MAX(fi(a1),a [3365]
3)-1)-a3/100:d(a1,a2)=0:GOTO 1690
1550 IF k(a1)+l(a(0))<a4 THEN a4=k(a1)+l(a [1716]
(0))
1560 j(a1,0)=j(a1,0)-a4*b(a(0),a3):j(a1,a3 [8129]
)=j(a1,a3)+a4:k(a1)=k(a1)-a4:IF k(a1)<0 TH
EN l(a(0))=l(a(0))+k(a1):j(a1,a3)=j(a1,a3)
+k(a1):jb(a(0),a3)=jb(a(0),a3)-k(a1):k(a1)
=0
1570 d(a1,a2)=0:GOSUB 500:n(a(0),7)=n(a(0) [6427]
,7)+"Kauf von"+STR$(a4)+" T "+m(a3):GOTO 1
530
1580 IF j(a1,a3)+jb(a(0),a3)<a4 THEN a4=j( [2420]
a1,a3)+jb(a(0),a3)
1590 j(a1,0)=j(a1,0)+a4*b(a(0),a3):j(a1,a3 [6660]
)=j(a1,a3)-a4:k(a1)=k(a1)+a4:a5=j(a1,a3):I
F a5<0 THEN jb(a(0),a3)=jb(a(0),a3)+a5:l(a
(0))=l(a(0))+a5:k(a1)=k(a1)-a5:j(a1,a3)=0
1600 d(a1,a2)=0:GOSUB 500:n(a(0),7)=n(a(0) [4087]
,7)+"Verkauf von"+STR$(a4)+" T "+m(a3):GOT
O 1530
1610 i(a1)=0:d(a1,a2)=0:GOSUB 500:n(a(0),7 [6797]
)=n(a(0),7)+"Schiff "+s(a1)+" reparieren."
:GOTO 1530
1620 IF a(0)=fi(a3) THEN d(a1,a2)=0:GOTO 1 [1560]
530 ELSE 1690
1630 IF a4>jb(a(0),a3) THEN a4=jb(a(0),a3) [2729]
1640 IF a4>k(a1) THEN a4=k(a1)             [522]
1650 jb(a(0),a3)=jb(a(0),a3)-a4:j(a1,a3)=j [10044]
(a1,a3)+a4:l(a(0))=l(a(0))+a4:k(a1)=k(a1)-
a4:GOSUB 500:n(a(0),7)=n(a(0),7)+STR$(a4)+
" T "+m(a3)+" aufs Schiff":d(a1,a2)=0:GOTO
 1530
1660 IF a4>j(a1,a3) THEN a4=j(a1,a3)       [1324]
1670 IF a4>l(a(0)) THEN a4=l(a(0))         [1713]
1680 jb(a(0),a3)=jb(a(0),a3)+a4:j(a1,a3)=j [10028]
(a1,a3)-a4:l(a(0))=l(a(0))-a4:k(a1)=k(a1)+
a4:GOSUB 500:n(a(0),7)=n(a(0),7)+STR$(a4)+
" T "+m(a3)+" ins Lager":d(a1,a2)=0:GOTO 1
530
1690 NEXT a1:IF INT(z/5)<>z/5 THEN 1710    [2536]
1700 FOR a1=1 TO 10:a3=ROUND(1.5+RND,1):FO [5511]
```

```
R a2=1 TO 10:b(a1,a2)=INT(b(a1,a2)*a3):NEX
T a2,a1
1710 GOSUB 370:GOSUB 400:IF z>jl THEN t="I [5288]
hre Lebenszeit ist abgelaufen.":GOTO 2180
ELSE RETURN
1720 '                                     [117]
1730 ' Handelspartner                      [1221]
1740 '                                     [117]
1750 CLS#4:GOSUB 270:CLS#1:GOSUB 420:IF PO [11876]
S(#0)<40 THEN PRINT#3,"Es ist kein Schiff
im Hafen,":PRINT#3,"mit dem Sie reisen koe
nnten !":PRINT#3,"-Warte auf Eingabe-":GOT
O 890
1760 PRINT#3,"Bitte waehlen Sie das Schiff [6348]
,":PRINT#3,"mit dem Sie mitfahren wollen."
:GOSUB 710:IF g>0 THEN 180
1770 IF d(a,1)>0 THEN PRINT#3,"Fuer dieses [5532]
 Schiff ist ein":PRINT#3,"Fahrplan aufgest
ellt.":GOTO 890
1780 PRINT#3,"Sie fahren mit dem Schiff:": [12728]
PRINT#3,s(a):a(1)=a:GOSUB 530:PRINT#3,"Beh
alten Sie folgendes im Auge:":PRINT#3,"Zei
t, Abnuetzung des Schiffes, Ort":a(0)=0:f=
1:CALL &8F25
1790 DI:LOCATE 41,17:PRINT SPACE$(30):LOCA [19178]
TE 41,19:PRINT SPACE$(30):LOCATE 1,17:PRIN
T"Monatsablauf in Prozent":PLOT 320,126,1:
DRAWR 202,0:DRAWR 0,18:DRAWR -202,0:DRAWR
0,-18:PRINT:PRINT"Abnuetzung des Schiffes
in Prozent":PLOT 320,94:DRAWR 202,0
1800 DRAWR 0,18:DRAWR -202,0:DRAWR 0,-18:F [15726]
OR a1=0 TO i(a(1)) STEP 0.5:PLOT 320+a1*2,
94:DRAWR 0,18:NEXT:FOR a1=0 TO a(0):PLOT 3
20+a1*2,126:DRAWR 0,18:NEXT:WHILE INKEY$<>
"":WEND:GOSUB 650:a1=REMAIN(0):EVERY 10,0
GOSUB 670:GOSUB 570
1810 IF g>0 THEN a1=REMAIN(0):PLOT x,y,0:P [9549]
LOTR 1,0:f=0:fi(a(1))=1:CALL &8F25:GOTO 18
0 ELSE EI:t=INKEY$:IF t<CHR$(240) THEN 181
0 ELSE IF l>0 THEN l=0:GOSUB 330:GOSUB 400
:GOSUB 370
1820 DI:a(0)=a(0)+4:GOSUB 570:PLOT x,y,1:P [4805]
LOTR 1,0:ON ASC(t)-239 GOTO 1830,1840,1850
,1860
1830 y=y+2:IF TEST(x,y)=1 THEN 1870 ELSE P [3791]
LOT x,y-2,0:PLOTR 1,0:GOTO 1810
1840 y=y-2:IF TEST(x,y)=1 THEN 1870 ELSE P [4866]
LOT x,y+2,0:PLOTR 1,0:GOTO 1810
1850 x=x-2:IF TEST(x,y)=1 THEN 1870 ELSE P [3606]
LOT x+2,y,0:PLOTR 1,0:GOTO 1810
1860 x=x+2:IF TEST(x,y)=1 THEN 1870 ELSE P [3979]
LOT x-2,y,0:PLOTR 1,0:GOTO 1810
1870 IF x=318 THEN x=636:PLOT 320,y,0:PLOT [4720]
R 1,0:GOTO 1810 ELSE IF x=638 THEN x=320:P
LOT 636,y,0:PLOTR 1,0:GOTO 1810
1880 RESTORE 3070:l=0:FOR a1=1 TO 10:READ  [6933]
a2,a3,a4,a5:IF x>a2 AND x<a3 THEN IF y>a4
AND y<a5 THEN l=a1:a1=10:GOSUB 330
1890 NEXT:PRINT#3,"Sie sind in "l$(l):x=x- [13421]
2*(t=CHR$(242))+2*(t=CHR$(243)):y=y+2*(t=C
HR$(240))-2*(t=CHR$(241)):EI:IF l=0 THEN 1
930 ELSE fi(a(1))=1:GOSUB 400:GOSUB 330:GO
SUB 370:IF l(l)<0 THEN 1950
1900 WHILE INKEY$<>"":WEND:PRINT#3,"Sie ko [16183]
ennen Geschaefte abwickeln.":LOCATE 1,20:P
RINT"Die letzten Taetigkeiten:":CLS#6:FOR
a=1 TO 7:IF n(l,a)<>"" THEN PRINT#6,LEFT$(
n(l,a),4);RIGHT$(n(l,a),LEN(n(l,a))-7)
1910 NEXT:WHILE INKEY$<CHR$(240) AND g=0:W [4817]
END:LOCATE 1,20:PRINT SPACE$(27):CLS#6:GOS
UB 650:GOTO 1810
1920 DI:RESTORE 3080:WHILE t<>LEFT$(l$(l), [7187]
2):READ t:WEND:READ a1,a2:PLOT a1,a2,1:REA
D a1,a2:WHILE a1<>0 OR a2<>0:DRAWR a1,a2:R
EAD a1,a2:WEND:l(l)=0:RETURN
1930 PRINT#3,"Wollen Sie das Schiff repari [8251]
eren ?":GOSUB 620:IF t="n" THEN GOSUB 650:
GOTO 1810 ELSE PRINT#3,"Abnuetzung:"i(a(1)
)
1940 INPUT#3,"Reparatur in % ";a3:IF a3>i( [8957]
a(1)) THEN 1940 ELSE a(0)=a(0)+INT(a3/4)*4
:i(a(1))=i(a(1))-a3:PRINT#3,"Die Reparatur
 dauerte"INT(a3*0.3)"Tage":GOSUB 570:GOTO
1790
1950 IF l(l)<-1 THEN PRINT#3,"Es braucht n [15914]
och Zeit, bis Gras ueber":PRINT#3,"die alt
e Geschichte gewachsen ist.":GOTO 1810 ELS
E a1=REMAIN(0):PRINT#3,"Waehlen Sie die Ge
schenke aus.":a(2)=0
1960 GOSUB 450:IF a2>j(a(1),a3) THEN PRINT [16081]
#3,"So viel ist nicht vorhanden.":PRINT#3,
"Waehlen Sie neu!":GOTO 1960 ELSE IF a2=0
THEN PRINT#3,"Kein Geschenk registriert."
ELSE j(a(1),a3)=j(a(1),a3)-a2:LOCATE 1,13:
PRINT SPACE$(160):k(a(1))=k(a(1))+a2
1970 a(2)=a(2)+a2*b(1,a3):a=a(1):GOSUB 530 [6324]
:GOSUB 370:PRINT#3,"Weitere Geschenke ausw
aehlen ?":GOSUB 620:IF t="j" THEN GOSUB 65
0:GOTO 1960 ELSE EVERY 10,0 GOSUB 670
1980 IF a(2)<81 THEN PRINT#3,"Das Land hat [11135]
 mehr Geschenke er-":PRINT#3,"wartet. -Bes
uchsverbot fuer 3 Jahre-":l(l)=l(l)-36:GOS
UB 690:n$(l,7)=n(l,7)+"Besuchsverbot fuer
3 Jahre":GOTO 1810
1990 PRINT#3,"Sie haben einen neuen Handel [11117]
spartner":PRINT#3,"gefunden.":l(l)=0:GOSUB
 1920:GOSUB 690:GOSUB 400:n(l,7)=n(l,7)+"L
and wird Handelspartner.":GOTO 1930
2000 '                                     [117]
2010 ' Warentransfer                       [1607]
2020 '                                     [117]
2030 IF l=0 THEN PRINT#3,"An diesem Ort ni [9358]
cht moeglich!":GOTO 890 ELSE IF l(l)<0 THE
N PRINT#3,"Hier ist kein Handelsstuetzpunk
t.":GOTO 890
2040 GOSUB 270:CLS#1:GOSUB 420:PRINT#3,"Wa [7895]
ehlen Sie das betreffende Schiff":GOSUB 86
0:GOSUB 710:IF g>0 THEN 180
2050 PRINT#3,"Geben Sie den Artikel ein":G [10213]
OSUB 780:PRINT#3,"1: Transfer von Schiff n
ach Lager":PRINT#3,"2: Transfer von Lager
nach Schiff":IF g>0 THEN 180
2060 t=INKEY$:IF t<>"1" THEN 2100 ELSE IF  [11347]
l(l)=0 THEN PRINT#3,"Kein Lagerplatz vorha
nden.":GOTO 2140 ELSE IF j(a,a1)=0 THEN PR
INT#3,"Kein "+m(a1)+" auf Schiff vorhanden
.":GOTO 2140
2070 INPUT#3,"Menge";a2:IF a2>j(a,a1) OR a [8572]
2>l(l) THEN 2070 ELSE IF a2=0 THEN PRINT#3
,"Kein Transfer !!!":GOTO 890
2080 GOSUB 690:n(l,7)=n(l,7)+STR$(a2)+" T  [6305]
"+m(a1)+" ins Lager":j(a,a1)=j(a,a1)-a2:jb
(l,a1)=jb(l,a1)+a2
2090 k(a)=k(a)+a2:l(l)=l(l)-a2:GOSUB 400:G [4092]
OSUB 370:GOSUB 830:GOTO 2140
2100 IF t<>"2" THEN 2060 ELSE IF jb(l,a1)= [11709]
0 THEN PRINT#3,"Kein "+m(a1)+" im Lager vo
rhanden.":GOTO 2140 ELSE IF k(a)=0 THEN PR
INT#3,"Kein freier Schiffsraum!":GOTO 2140
2110 INPUT#3,"Menge";a2:IF a2>jb(l,a1) OR  [6752]
a2>k(a) THEN 2110 ELSE IF a2=0 THEN PRINT#
3,"Kein Transfer !!!":GOTO 890
2120 GOSUB 690:n(l,7)=n(l,7)+STR$(a2)+" T  [6072]
"+m(a1)+" aufs Schiff":jb(l,a1)=jb(l,a1)-a
2:j(a,a1)=j(a,a1)+a2
2130 k(a)=k(a)-a2:l(l)=l(l)+a2:GOSUB 400:G [5202]
OSUB 370:GOSUB 830:GOTO 2140
2140 PRINT#3,"Transfer beendet.":GOTO 890  [2535]
2150 '                                     [117]
2160 ' Spielende                           [324]
2170 '                                     [117]
2180 LOCATE 40-INT(LEN(t)/2),14:PRINT CHR$ [3662]
(24)+" "+t+" "
2190 LOCATE 35,3:PRINT CHR$(24)+" SPIELEND [2768]
E "+CHR$(24)
2200 LOCATE 33,3:PRINT" SPIELENDE "+CHR$(2 [4491]
4)+CHR$(7):WHILE INKEY$<>"":WEND:LOCATE 1,
1:MEMORY &AB7F:END
2210 '                                     [117]
2220 ' Wareneinkauf                        [521]
2230 '                                     [117]
2240 IF l=0 THEN PRINT#3,"Kein Handel auf  [4228]
dem Ozean moeglich!":GOTO 890
2250 IF l(l)<0 THEN 890                    [185]
2260 IF ab+l(l)=0 THEN PRINT#3,"Kein Lager [15018]
platz vorhanden.":GOTO 890 ELSE GOSUB 910:
PRINT#3,"Waehlen Sie den Artikel.":g=0:CLS
#4:GOSUB 780:a5=a1:IF g>0 THEN 180 ELSE IN
PUT#3,"Menge";a2:e=e-a2*b(l,a1):GOSUB 350:
GOSUB 860:CLS#4
2270 IF a2=0 THEN PRINT#3,"Kein Einkauf !! [3090]
!":GOTO 890
2280 GOSUB 690:n(l,7)=n(l,7)+"Kauf von"+ST [4441]
R$(a2)+" T "+m(a5)
2290 PRINT#3,"Sie muessen";STR$(a2)"T unte [10295]
rbringen.":IF l(l)>0 THEN PRINT#3,"Moechte
n Sie das Lager benutzen":GOSUB 620:a=0:IF
 t="j" THEN 2310
```

LISTINGS

LISTINGS

```
2300 PRINT#3,"Verteilen Sie den Einkauf.":  [7510]
PRINT#3,"Waehlen Sie die Schiffe aus.":GOS
UB 710:IF g>0 THEN 180
2310 INPUT#3,"Teilmenge";a3:IF a3>a2 THEN   [7135]
PRINT#3,"So viel ist nicht vorhanden.":GOT
O 2310
2320 IF a=0 THEN IF l(1)<a3 THEN PRINT#3,"  [6813]
Achten Sie auf den Lagerraum.":GOTO 2310 E
LSE l(1)=l(1)-a3:jb(1,a5)=jb(1,a5)+a3:GOTO
 2340
2330 IF k(a)<a3 THEN PRINT#3,"Achten Sie a  [7937]
uf die Ladefaehigkeit.":GOTO 2310 ELSE k(a
)=k(a)-a3:j(a,a5)=j(a,a5)+a3:GOTO 2340
2340 a2=a2-a3:GOSUB 860:GOSUB 400:GOSUB 37  [3638]
0:IF ab+l(1)>=1 AND a2>0 THEN 2290
2350 IF a2=0 THEN 890 ELSE PRINT#3,"Kein L  [11236]
agerplatz mehr vorhanden.":PRINT#3,"Sie mu
ssten"+STR$(a2)+"T "+m(a5)+" verschenken."
:PRINT#3,"Verlust:"a2*b(1,a5):GOTO 890
2360 '                                      [117]
2370 ' Warenverkauf                         [1053]
2380 '                                      [117]
2390 IF l(1)<0 THEN 890                     [185]
2400 GOSUB 910:PRINT#3,"Wollen Sie Schiffs  [11735]
ladung verkaufen":GOSUB 620:a=0:g=0:IF t="
j" THEN PRINT#3,"Waehlen Sie das entsprech
ende Schiff":CLS#4:GOSUB 710
2410 IF g>0 THEN 180 ELSE CLS#4:GOSUB 780:  [1736]
IF g>0 THEN 180
2420 INPUT#3,"Verkauf";a5:IF a5=0 THEN PRI  [4023]
NT#3,"Kein Verkauf !!!":GOTO 890
2430 IF a>0 THEN 2440 ELSE IF a5>jb(1,a1)   [9877]
THEN PRINT#3,"Diese Menge ist nicht im Lag
er":GOTO 2420 ELSE l(1)=l(1)+a5:jb(1,a1)=j
b(1,a1)-a5:GOTO 2450
2440 IF j(a,a1)<a5 THEN PRINT#3,"Diese Men  [7399]
ge ist nicht auf dem Schiff":GOTO 2420 ELS
E k(a)=k(a)+a5:j(a,a1)=j(a,a1)-a5
2450 e=e+b(1,a1)*a5:GOSUB 690:n(1,7)=n(1,7  [8216]
)+"Verkauf von"+STR$(a5)+" T "+m(a1):GOSUB
 350:GOSUB 370:GOSUB 400:CLS#4:GOTO 890
2460 '                                      [117]
2470 ' Lager bauen                          [351]
2480 '                                      [117]
2490 IF l(1)<0 THEN 890                     [185]
2500 WHILE INKEY$<>"":WEND:INPUT#3,"Aufnah  [9017]
mefaehigkeit";a1:INPUT#3,"Bauzeit in Monat
en";a2
2510 PRINT#3,"Wollen Sie neue Eingaben mac  [10011]
hen":GOSUB 620:IF t="j" THEN 2490 ELSE PRI
NT#3,"Moechten Sie einen Kostenvoranschlag
":GOSUB 620:IF t="n" THEN 2530
2520 a3=INT((a1*18+a1^2/a2)*(0.9+RND/5)):P  [9432]
RINT#3,"Ein unabhaengiger Schaetzer:"+STR$
(a3):GOTO 2510
2530 PRINT#3,"Aufnahmefaehigkeit:"+STR$(a1  [10925]
):PRINT#3,"Bauzeit:"+STR$(a2)+" Monate":PR
INT#3,"Kann die Errichtung beginnen.":GOSU
B 620:IF t="n" THEN 890
2540 GOSUB 270:CLS#1:a3=a1/4*b(1,4):PRINT#  [17109]
1,"Sie benoetigen"+STR$(a1/4)+" T Stahl:"+
STR$(a3):e=e-a3:a3=a1/4*b(1,3):PRINT#1,"Si
e benoetigen"+STR$(a1/4)+" T Holz:"+STR$(a
3):e=e-a3:a3=a1*11:PRINT#1,"Sie benoetigen
"+STR$(a1*7)+" qum Flaeche:"+STR$(a3):e=e-
a3
2550 GOSUB 350:a3=a1*10:PRINT#1,"Arbeiterl  [13834]
oehne:"+STR$(a3):e=e-a3:GOSUB 350:ll(1)=a2
+a1/10000:a3=z:z=z+a2/100:GOSUB 300:PRINT#
1,"Bauende: "+q((z-INT(z))*100)+STR$(INT(z
)):z=a3:GOSUB 300:GOSUB 650:g=0:WHILE g=0:
WEND:GOTO 180
2560 '                                      [117]
2570 ' Schiffe planen                       [1519]
2580 '                                      [117]
2590 IF l(1)<0 THEN PRINT#3,"Hier ist kein  [3355]
 sicherer Ort.":GOTO 890 ELSE IF f(1)<>0 T
HEN 2620
2600 PRINT#3,"In diesem Hafen besitzen Sie  [16027]
 keine":PRINT#3,"Reederei. Sie koennten ei
ne fuer":PRINT#3,"25000 kaufen.":PRINT#3,"
Kaufen Sie (j/n)":GOSUB 620:IF t="j" THEN
PRINT#3,"Sind Sie sicher":GOSUB 620
2610 IF t="j" THEN e=e-25000:GOSUB 350:f(1  [5763]
)=-1:GOSUB 690:n(1,7)=n(1,7)+"Reederei fue
r 25000 kaufen" ELSE 2640
2620 IF f(1)<0 THEN 2640 ELSE GOSUB 270:CL  [13142]
S#1:LOCATE 40,1:PRINT"Es wird gerade gearb
eitet:":a3=z:z=z+INT(f(1))/100:GOSUB 300:L
OCATE 40,3:PRINT"Bauende: "q((z-INT(z))*10
0)+STR$(INT(z)):z=a3:GOSUB 300:LOCATE 40,4
2630 PRINT"Ladefaehigkeit:"+STR$(ROUND((f(  [6763]
1)-INT(f(1)))*10000)):LOCATE 40,5:PRINT"Sc
hiffsname: ";t2(1)
2640 CLS#4:LOCATE 1,13:PRINT"Es wird auf d  [6866]
em schwarzen Schiffsmarkt angeboten:":PRIN
T:
2650 ac=l*INT(z):FOR a=1 TO 5:PRINT STR$(a  [11565]
)+".   Ladefaehigkeit"+STR$(ac MOD (100+a)
)+" Tonen, Abnuetzung"+STR$(ac/a MOD (30+a
)+1)+" %, Preis:"+STR$(INT((ac MOD (100+a)
)/(ac/a MOD (30+a)+1)^0.1*130)):NEXT
2660 LOCATE 1,25:PRINT CHR$(24)+"1) Schiff  [6920]
 kaufen . 2) Schiff verkaufen . 3) Schiff
reparieren . 4) Schiff bauen"+CHR$(24);:g=
0
2670 PRINT#3,"Bitte waehlen Sie"            [1760]
2680 IF g>0 THEN 180 ELSE t=INKEY$:IF t<"1  [4700]
" OR t>"4" THEN 2680 ELSE ON ASC(t)-48 GOT
O 2690,2720,2770,2800
2690 INPUT#3,"Welches";a2:IF a2<1 OR a2>5   [2723]
THEN 2670
2700 e=e-INT((ac MOD (100+a2))/(ac/a2 MOD   [11491]
(30+a2)+1)^0.1*130):GOSUB 350:h=h+1:fi(h)=
1:k(h)=ac MOD (100+a2):i(h)=ac/a2 MOD (30+
a2)+1:INPUT#3,"Schiffsname";t:s(h)=LEFT$(t
,10):FOR a=1 TO 10:j(h,a)=0:NEXT
2710 GOSUB 270:CLS#1:GOSUB 420:GOSUB 370:G  [3278]
OTO 2670
2720 PRINT#3,"Welches Schiff wollen Sie ve  [6958]
rkaufen?":INPUT#3,t:a1=0:FOR a=1 TO h:IF L
OWER$(s(a))=LOWER$(t) THEN a1=a:a=h
2730 NEXT:IF a1=0 THEN PRINT#3,"Dieses Sch  [7554]
iff kenne ich nicht!":GOTO 2670 ELSE IF fi
(a1)<>1 THEN PRINT#3,"Das Schiff ist nicht
 im Hafen.":GOTO 2670
2740 a2=k(a1)*90:FOR a=1 TO 10:a2=a2+j(a1,  [11966]
a)*(90+b(1,a)):NEXT:a2=INT(a2*(1-i(a1)/200
)):PRINT#3,"Es wird Ihnen"+STR$(a2)" gebot
en.":PRINT#3,"Verkaufen Sie jetzt auch noc
h ?":GOSUB 620:IF t="n" THEN 2670
2750 e=e+a2:GOSUB 690:n(1,7)=n(1,7)+s(a1)+  [11741]
" verkaufen fuer"+STR$(a2):FOR a=a1 TO h:f
i(a)=fi(a+1):FOR a4=1 TO 10:j(a,a4)=j(a+1,
a4):NEXT:i(a)=i(a+1):k(a)=k(a+1):s(a)=s(a+
1):NEXT
2760 h=h-1:GOSUB 350:GOSUB 370:GOSUB 270:C  [3207]
LS#1:GOSUB 420:GOTO 2670
2770 IF h=0 THEN PRINT"Sie besitzen keine   [3724]
Schiffe.":GOTO 2670
2780 GOSUB 710:IF g>0 THEN 180 ELSE PRINT#  [9313]
3,"Abnuetzung:"+STR$(i(a)):INPUT#3,"Repara
tur: ",a1:IF a1>i(a) THEN PRINT#3,"Keine R
eparatur":GOTO 2640
2790 i(a)=i(a)-a1:e=e-a1*9:PRINT#3,"Die Re  [5323]
paratur kostete"+STR$(a1*9):GOSUB 350:GOTO
 2640
2800 IF f(1)=0 THEN PRINT#3,"Sie besitzen   [9071]
keine Reederei.":GOTO 2670 ELSE IF f(1)>0
THEN PRINT#3,"Es wird gerade ein Schiff ge
baut.":GOTO 2670
2810 INPUT#3,"Ladefaehigkeit";a1:INPUT#3,"  [3197]
Bauzeit";a2
2820 a3=1+ABS(a1-10*a2)/100:IF a1-10*a2>0   [2552]
THEN a3=a3*2
2830 a3=a3*50*a1:PRINT#3,"Kostenvoranschla  [6732]
g:"+STR$(a3):PRINT#3,"Neue Eingaben":GOSUB
 620:IF t="j" THEN 2810
2840 PRINT#3,"Soll der Bau beginnen ?":GOS  [7353]
UB 620:IF t="n" THEN 2670 ELSE INPUT#3,"Sc
hiffsname";t:t2(1)=LEFT$(t,10)
2850 e=e-a3:f(1)=a2+a1/10000:GOSUB 350:GOT  [3216]
O 2620
2860 '                                      [117]
2870 ' Data's                               [368]
2880 '                                      [117]
2890 DATA Januar,Februar,Maerz,April,Mai,J  [17844]
uni,Juli,August,September,Oktober,November
,Dezember,Gewuerze,Deutschland,Seide,Portu
gal,Getreide,Elfenbeinkueste,Holz,Madagask
ar,Stahl,Indien,Salz,China,Kaffee,Australi
en,Tabak,Brasilien,Reis,Mexiko,Porzellan,U
SA
2900 DATA unbekannt                         [1129]
2910 REM ... fuer Anfangspreise             [887]
2920 DATA 27,28,27,18,19,24,25,37,45,21     [1353]
2930 DATA 25,33,32,26,23,24,20,29,44,30     [2149]
2940 DATA 26,38,37,23,28,28,18,17,32,28     [1877]
```

```
2950 DATA 23,30,35,26,31,33,19,19,29,30          [1807]
2960 DATA 21,26,29,29,30,26,22,21,23,32          [1151]
2970 DATA 25,21,34,25,31,24,26,30,19,22          [1679]
2980 DATA 26,27,27,31,28,19,23,26,38,41          [1619]
2990 DATA 29,33,30,16,36,28,23,18,44,37          [1703]
3000 DATA 16,39,32,29,33,23,20,24,39,32          [1773]
3010 DATA 27,27,22,26,20,22,27,26,30,29          [1365]
3020 REM ... fuer Entfernungen                   [687]
3030 DATA 1,2,4,6,8,8,3,3,3,1,3,5,7,7,2,2,       [3775]
2,2,4,6,6,2,1,2,2,3,3,4,3,5,2,2,6,5,7,3,2,
8,2,3,5,4,2,1,3
3040 REM ...fuer Tastenbelegung                  [1753]
3050 DATA 61,15,27,13,58,14,38,5,35,20,62,       [2897]
12,69,4,54,10,38,11,42,3
3060 REM ... fuer neuen Handelspartner           [1640]
3070 DATA 470,474,366,370,454,458,348,356,       [7937]
452,464,308,312,514,526,262,280,532,556,31
0,334,576,592,330,362,578,622,246,286,408,
432,260,310,346,376,326,346,342,410,344,36
2
3080 DATA De,473,366,0,-4,-3,0,0,4,1,0,0,-       [10717]
4,1,0,0,4,0,0,Po,457,354,0,-4,1,0,0,4,0,0,
El,454,310,8,0,0,2,-8,0,0,2,8,0,0,2,-8,0,0
,2,8,0,0,0,Ma,518,266,0,6,8,8,0,0,In,542,3
42,0,-2,2,0,-2,-2,12,0,0,-2,-14,0,-2,-2,16
,0,0,-2,-12,0,0,-2,10
3090 DATA 0,0,-2,-10,0,2,-2,6,0,-2,-2,-4,0 [10850]
,0,-2,2,0,0,-6,1,0,0,6,2,-2,0,0,Ch,568,372
,-2,-2,4,0,4,-2,-10,0,-2,-2,18,0,2,-2,-22,
0,-2,-2,32,0,0,-2,-34,-0,-4,-2,36,0,0,-2,-
46,0,2,2,2,0,-6,2,2,-6,38,0,0,-2,-36,0,0,-
2,40,0,0,-2,-38,0,0,-2,40,0,0,-2,-38,0,0
3100 DATA -2,42,0,0,-2,-34,0,4,-2,32,0,-2,  [11155]
-2,-24,0,22,-2,0,0,Au,600,250,12,0,2,2,-14
,0,0,2,16,0,2,2,-38,0,0,2,40,0,0,2,-40,0,0
,2,40,0,0,2,-40,0,0,2,40,0,0,2,-40,0,0,2,4
0,0,-2,2,-32,0,2,2,28,0,0,2,-24,0,0,2,24,0
,-10,0,0,2,-4,0,2,2,2,0,0,0
3110 DATA Br,414,262,2,0,0,2,-4,0,-2,2,6,0 [11075]
,2,2,-12,0,-2,2,14,0,2,2,-14,0,-2,2,20,0,0
,2,-22,0,0,2,24,0,0,2,-24,0,-2,2,26,0,0,2,
-28,0,0,2,28,0,2,2,-32,0,-2,2,36,0,0,2,-36
,0,0,2,36,0,-6,2,-30,0,0,2,20,0,-2,2,-18,0
,2,2,14,0,0,2,-10,0,4,2,6,0,0,0
3120 DATA Me,356,328,18,0,0,2,-20,0,0,2,12 [7729]
,0,0,2,-12,0,0,2,12,0,0,2,-12,0,-2,2,16,0,
0,2,-18,0,0,2,26,0,0,0,US,346,346,38,0,2,2
,-42,0,0,2,46,0,4,2,-48,0,0,2,48,0,2,2,-50
,0,2,2,50,0,2,2,-52,0,0,0
```

fang wählen Sie das Schiff aus, mit dem Sie mitfahren wollen. Es ist zu beachten, daß dieses Schiff noch keinen Fahrplan hat. Alles O.K. - so finden Sie rechts oben eine Weltkarte, unten die Anzeigen: Monatsablauf und die Abnützung. Beim Monatsablauf ist zu erwähnen, daß manchmal bei einem Monatswechsel eine kurze Zeitverzögerung (max. 1 Sekunde) auftritt. Die Abnützung darf niemals 65 % erreichen, da sonst das Schiff sinkt. In der Weltkarte blinkt ein kleiner Punkt, der mit den Cursortasten gesteuert wird (COPY besitzt keine Funktion). Sollten Sie an einem Land anstoßen (meistens unbekannt), so haben Sie Gelegenheit, das Schiff zu reparieren (Eingabewindow beobachten). Stoßen Sie auf eines der zehn bestimmten Länder (siehe Schiff senden) und Sie haben es bereits entdeckt, so erscheint im unteren Teil des Bildschirms eine Auflistung der Tätigkeiten des Landes. Anderenfalls werden Sie aufgefordert, Geschenke auszuwählen (Mittelwert 4 - 6 Tonnen).

Haben Sie zu wenig Geschenke ausgewählt, erhalten Sie ein Besuchsverbot für drei Jahre. Hatten Sie Erfolg, so wird die Fläche des betreffenden Landes ausgefüllt und Ihnen die Möglichkeit gegeben, das Schiff zu reparieren. Nun können Sie wieder nach Herzenslust durch die Weltmeere fahren...

Wenn Sie bei der Anzeige Ort nicht "unbekannt" lesen, so können Sie jederzeit das Hilfsmenue voll ausschöpfen.

(Richard Krampfel)

Sie wollen uns ein Programm zusenden? Folgende Regeln sollten Sie beachten:

1) Schicken Sie Ihre Programme auf Kassette oder Diskette. Dabei mehrmals hintereinander abspeichern.

2) Legen Sie eine ausführliche Programmbeschreibung bei, aus der eindeutig der Sinn und Zweck des Programmes hervorgeht. Eine Variablenliste ist sehr wichtig, in ihr sollten alle im Programm definierten Variablen mit ihrer Verwendung aufgeführt sein.
Beispiel: n$ = Nachname
g = glücklicher usw.

3) Wenn Sie gleichzeitig Besitzer eines Druckers sind, schicken Sie auf jeden Fall ein Programmlisting mit.

4) Vergessen Sie Ihren Absender nicht! Vollständige Anschrift, wenn möglich auch die Telefonnummer, ersparen unnötige Verzögerungen bei eventuellen Nachfragen unsererseits.

5) Behalten Sie auf jeden Fall eine Kopie Ihres Programmes, unvorhersehbare Umstände, wie z.B. Bandriß der Kassette, werden somit nicht zur Katastrophe.

6) Wenn Sie sogar mehrere Programme geschrieben haben und diese veröffentlichen wollen, nehmen Sie bitte für jedes Programm eine separate Kassette (mehrmals abspeichern!).

Damit erleichtern Sie unsere Arbeit wesentlich und eine schnellere Bearbeitung wird möglich.

7) Versehen Sie das Programm mit den sehr wichtigen REM-Statements. Das fördert nicht nur die Übersicht für uns und unsere Leser, sondern unterstützt zusätzlich das strukturierte Programmieren. Merke: Auch erfahrene Programmierer verwenden REM-Zeilen, das Programm sollte schließlich jeder verstehen und anwenden können.

8) Vermeiden Sie möglichst Variablennamen, wo leicht l und i oder 0 und o verwechselt werden können. Variablen sollten grundsätzlich in Kleinbuchstaben geschrieben werden und möglichst verständlich sein. Die Variable butter sagt mehr aus, als nur bu.

9) Bieten Sie das Programm niemals verschiedenen Verlagen gleichzeitig an! Mit der Veröffentlichung und dem daraus resultierenden Honorar, gehen die Urheberrechte an den Verlag über! Sollte Ihr Programm dann noch in einer anderen Zeitschrift abgedruckt werden, liegt ein Verstoß gegen das Wettbewerbsrecht vor.

10) Jedes bei uns ordnungsgemäß eingegangene Programm wird ausführlich begutachtet und getestet. Sie erhalten von uns dann umgehend Bescheid, ob Ihr Programm veröffentlicht wird oder nicht. Wenn Sie das Programm nach unserer Begutachtung zurückhaben wollen, legen Sie bitte der Einsendung einen frankierten und adressierten Rückumschlag bei. Die Kosten einer eventuellen Rücksendung sind in jedem Fall vom Einsender zu tragen.

Falls wir uns für den Abdruck Ihres Programmes entschieden haben, setzen wir uns unverzüglich mit Ihnen in Verbindung und besprechen Ihr verdientes Honorar!
Übrigens wäre es nett, wenn Sie uns auch ein paar persönliche Daten schreiben würden (z.B. Ihr Alter, Tätigkeit, wie und wann kam die Idee zu dem Programm, was alles machen Sie mit dem CPC usw.?). So, jetzt aber genug der Theorie, mit fast unerträglicher Spannung und Neugier wartet Ihre „Schneider CPC international" auf Ihre Beiträge.

Einsendungen bitte an: DMV - Daten & Medien Verlagsges.mbH, Fuldaer Straße 6, 3440 Eschwege

RSE Software

(0 23 05) 37 70

Die intelligente Dimension

SOFTWARE-AUTOREN GESUCHT!

Mathe-Pack
- Vom Anfänger bis zum Profi
- Eingebauter Taschenrechner
- Integrierte Hardcopyfunktion
- Algebra
- Geometrie
- Trigonometrie
- Mengenlehre
- Analysis
- Kurvendiskussion
- Benutzerfreundlich
- CPC 464, 664, 6128

Cassette 59,– DM*
Diskette 69,– DM*

Lagerverwaltung
- 100% Maschinencode
- Relative Dateiverwaltung
- Professioneller Editor
- Bestands-/Inventurliste
- Artikel VK-Preisliste
- Etikettenausdruck
- Ausdruck auf Einzel- o. Endlospapier
- Geschützter EK-Preis
- Statistik
- DIN-Tastatur
- Schnelle Sortierroutine
- Schnelles Auffinden von Daten
- 1350 Artikel je Datei möglich (beliebig viele Dateien)
- Bestell-Vorschlagsliste
- CPC 464, 664, 6128

Diskette 79,– DM*

Amsmonix
- Komfortabler Monitor-Debugger
- 100% Maschinencode
- RAM-ROM-AMSDOS-Monitor
- Disassembler
- Such-Füll-Modifizierfunktion
- Laden und Speichern von Files
- CAT-Funktion
- Intellig. Kopieren (Blockverschiebung)
- Registeranzeige
- Aufruf von Maschinenprogrammen
- Texteingabe
- Druckerprotokoll
- Anwahl auch von Expansion-Roms
- Monitor: ASCII o. Prüfsummen-Anz.
- CPC 464, 664, 6128

Cassette 49,– DM*
Diskette 59,– DM*

Adressverwaltung
- 100% Maschinencode
- Relative Dateiverwaltung
- Professioneller Editor
- Schnelle Sortierroutine
- Schnelles Auffinden von Daten
- 1 Benutzerfeld frei definierbar
- 900 Datensätze
- Etikettenausdruck für verschiedene Formate
- Listenausdruck mit Einzel- oder Endlosblattunterstützung
- Integrierte Hardcopyfunktion
- DIN-Tastaturbelegung
- CPC 464, 664, 6128

Diskette 59,– DM*

Disc-Scanner
- 100% Maschinencode
- Unterstützt 2 Laufwerke
- Disketten kopieren
- Lesen/Schreiben beliebiger Sektoren
- Sektoren modifizieren (Full Screen)
- Umrechnen von Blocks in Track/Sektor
- Files Umbenennen und Löschen
- Reaktivieren gelöschter Files
- Graph. Darstellung der Diskbelegung
- Usernummern wählbar
- Fileinformationen abrufen
- Formatieren / opt. Einzelspur
- Vendor / Data / 43 Tracks
- Integrierte Hardcopyfunktion
- CPC 464, 664, 6128

Diskette 79,– DM*

Disc-Sorter
- 100 % Maschinencode
- Relative Dateiverwaltung
- Automatische beidseitige Programmerfassung von einer Diskette in allen User-Bereichen
- wahlweise manuelle Erfassung
- Automatische Erfassung und Verarbeitung aller Fileinformationen
- Professioneller Editor
- schnelle Sortierroutine
- schnelles Auffinden von Daten
- Listenausdruck mit Einzel- oder Endlosblattunterstützung
- Integrierte Hardcopyfunktion
- CPC 464, 664, 6128

Diskette 69,– DM*

Intelligente Software für intelligente Anwender.

Disc-Scanner

Disc-gestattet Ihnen das Lesen und Beschreiben von beliebigen Sektoren Ihrer Diskette. Ein komfortabler Full-Screen-Editor ermöglicht das Modifizieren sämtlicher Bytes der zu bearbeitenden Diskette. Blocknummern können in Tracks und Sektoren umgerechnet werden, Files können umbenannt bzw. gelöscht werden, versehentlich gelöschte Files können wieder reaktiviert werden. Die Benutzung sämtlicher User-Bereiche ist möglich. Über jedes File, das sich auf einer Diskette befindet, ist die Abrufung aktueller Informationen möglich. Die Diskettenbelegung kann in einem Funktionsdiagramm dargestellt werden. Die sekundenschnelle Formatierroutine (17 Sek.) steht Ihnen im Vendor/Data und im Speziel-43-Track-Format zur Verfügung. Optimal formatiert Disc-Scanner auch Einzelspuren. Weiterhin ist ein Diskettenkopierprogramm sowie eine Hardcopyroutine integriert. Alle Diskettenoperationen können wahlweise mit einem oder zwei Laufwerken durchgeführt werden. Das Arbeiten mit Disc-Scanner ist ausgesprochen benutzerfreundlich, da durch sämtliche Programmpunkte per Menuesteuerung geführt wird. Dieses Programm ist in Maschinensprache geschrieben.

CPC 464, 664,6128,
Diskette 79,– DM

Neue Super-Programme in Vorbereitung ???

Super-Hardcopy
- 100% Maschinencode
- RSX-Befehle
- Menuegesteuert an versch. Druckertypen anpaßbar
- Text-Hardcopyfunktion
- Parameter wählbar
- Normal- und Inversdruck
- komprimierte Hardcopy
- 4-Farb-Darstellung möglich
- 1:1 Grafik-Hardcopy vom Bildschirm
- CPC 464, 664, 6128

Cassette 39,80 DM*
Diskette 49,80 DM*

Copy-Master
- Universelles File-Transfer Programm.
- 100% Maschinencode
- Kopiert und analysiert Ihre Software
- Cass. zu Cass.
- Cass. zu Disc.
- Disc. zu Cass.
- Disc. zu Disc.
- Disc Backup
- Diskettensicherheitskopie auf Cass. (kompl.)
- Programmresident im Speicher
- Integrierte Hardcopyfunktion
- Programmverschiebung bei AMSDOS-Überschreibung
- CPC 464, 664, 6128

Cassette 49,– DM*
Diskette 59,– DM*

Telecom 1000
- Professionelles DFÜ-Programm
- 100% Maschinencode
- Einstellen d. Übertragungsparameter
- 75, 110, 150, 300, 600, 1200 Baud
- Voll- und Halbduplex
- 5, 6, 7 oder 8 Datenbits
- 1 oder 2 Stopbits
- Keine, gerade oder ungerade Parität
- versch. Übertragungsprotokolle
- Über 32 K Textbuffer
- Notizzettel
- Professioneller Editor
- Übertragung von ASCII, Binär, Basic-Dateien
- Converterprogramm
- CPC 464, 664, 6128

Cassette 69,– DM*
Diskette 79,– DM*



Universaldatei
- 100% Maschinencode
- Relative Dateiverwaltung
- je nach Maske und Indexfelder mehrere Tausend Datensätze möglich
- bis zu 50 Felder pro Datensatz
- Frei definierbare Eingabemaske
- Such- und Druckermaske frei erstellbar
- Professioneller Editor
- Schnelle Sortierroutine
- Schnelles Auffinden von Daten
- DIN-Tastaturbelegung
- Integrierte Hardcopyfunktion
- CPC 464, 664, 6128

Diskette 79,– DM*

Vokabeltrainer
- Univers. f. jede Fremdsprache nutzbar
- 100 Vokabeln pro Unit
- unbegrenzte Unit-Bearbeitung
- 2 Lernstufen PP (Prägen/Prüfen)
- Einfache Handhabung durch Cursor-blocksteuerung
- Lernen von Mehrfachzuweisungen möglich (z.B. unregelmäßige Verben)
- Prozentuale Ergebnisauswertung, selbst bei Testunterbrechung
- Von Pädagogen empfohlen
- Deutscher Zeichensatz
- CPC 464, 664, 6128

Cassette 59,– DM*
Diskette 69,– DM*

Karteikasten
- 100 % Maschinencode
- Relative Dateiverwaltung
- 400 Karten pro Diskettenseite
- Beliebige Suchkriterien
- Professioneller Editor
- Schnelle Sortierroutine
- Schnelles Auffinden von Daten
- Integrierte Hardcopyfunktionen
- Listenausdruck
- DIN-Tastaturbelegung
- CPC 464, 664, 6128

Diskette 79,– DM*



R. Schuster Electronic

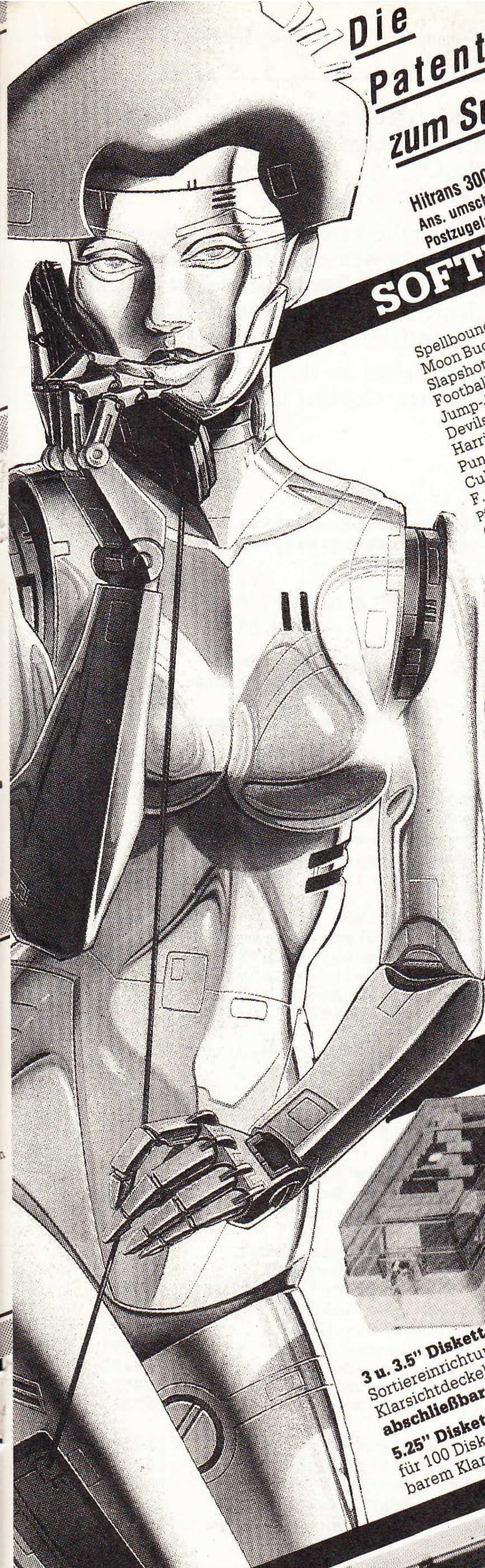
Schneider COMPUTER DIVISION Vertragshändler

OBERE MÜNSTERSTR. 33
☎ (0 23 05) 37 70
4620 CASTROP-RAUXEL

* UNVERBINDLICHE PREISEMPFEHLUNG

VERSAND PER NACHNAME ZUZÜGLICH VERSANDKOSTEN. INFO GEGEN 2,– DM IN BRIEFMARKEN.
HÄNDLERANFRAGEN ERWÜNSCHT.

RSE-SOFTWARE ERHALTEN SIE BEI IHREM FACHHÄNDLER ODER DIREKT BEI
Computer + Bürotechnik R. Kalisch Marktstr. 26, 4353 Oer-Erkenschwick
FERNSEH BERLET Elseyer Str. 12-14, 5800 Hagen-Hohenlimburg
Computerstudio HÖLSCHER Zeppelinstr. 7, 5010 Bergheim
SFK elektro GmbH Delsterner Str. 23, 5800 Hagen
GAI Computer Rathausstr. 28, 7107 Neckarsulm
und bei **R. SCHUSTER ELECTRONIC**



Die Patentlösung zum Superpreis!

Hitrans 300C 300 Bd. vollduplex, Orig./Ans. umschaltbar, Batterie- oder Netzbetrieb... Postzugelassen inkl. Netzteil .... DM **198,–**

SOFTWARE-FUNDGRUBE

	Cass.	Disk.
Spellbound	14.95	–
Moon Buggy	25,–	–
Slapshot	29,–	49,95
Football Manager	29,–	–
Jump-Jet	29,–	49,–
Devils Crown	29,–	49,–
Harrier Attack	29,50	59,–
Punchy	19,95	–
Cubit	29,50	–
F. Brunos Boxing	29,–	46,–
Ping Pong	27,–	
Ghostbusters	39,90	–
Exploding Fist	29,–	49,–
They sold a Million	29,–	49,–
They sold a Million II	39,95	59,95
V	34,95	–
Spindizzy	39,90	–
3 D Stunt Rider	24,95	–
Last V-8	14,95	–
Roland in der Zeit	39,50	–
Golf	39,50	–
Tennis	39,50	–
Pool Billard	39,50	49,–
Cyrus II Chess (3D)	29,–	44,95
Hi-Rise	–	39,95
Highway Encounter	–	39,–
Hyper-Sports	29,–	49,95
Sorcery	27,95	39,–
Wizard's Lair (zugl. C 64)	–	68,–
Super Sport I	–	87,–
Super Sport II	–	–
Winter Games	39,50	–
D. Thompsons Super Test	29,95	–
Yie Ar Kung-Fu	34,95	79,95
Elite	69,–	79,50
Super Games III	–	–
Airwolf	24,95	
Match Point	35,90	–
Bounty Bob Strikes Back	39,95	–
Spitfire 40	34,95	–
Selbstlernbasic I	79,50	–
Selbstlernbasic II	79,50	–
Super Games I	–	87,–
Super Games II	–	87,–
Computer Kurs	99,–	115,–
Texpack	–	198,–
Devpac	129,–	145,–
Hisoft Pascal	199,–	215,–
Word Star 3.0	–	199,–
Multiplan	–	199,–
dBase II	–	199,–
Compack Kompl.	–	798,–

Fragen Sie nach weiteren Programmen
Firmware Handbuch 89,–
Fachliteratur von Sybex und Markt & Technik

Jede Cassette 9,95

Finders Keepers
Locomotion
Nonterraqueous
Formula One Simulator
Vagan Attack
Soul of a Robot
Caves of Doom
Don't Panic
Forest at Worlds End
Tales of the Arabian Night
Hereos of Karn
One Man and his Droid
House of Usher (dtsch.)
Willow Pattern
Chiller
Chimera
Cylu
Manic Miner
Subsunk
Warlord
Into oblivion

HARDWARE

JOYSTICKS

Joystick-Verlängerungskabel 2 × 200 cm	24,90
Joystick-Adapter für Schneider/Amstrad zum Anschluß von 2 Standard-Joysticks	19,80
Quick Shot I	12,90
Quick Shot II	19,80
Quick Shot IV	24,90
Quick Shot V	27,95
Quick Shot VII	19,95
Joystick „The Stick", Einhand-Joystick, Steuerung durch die Handbewegung	49,–
Competition Pro mit Mikro-Schalter	39,80

DATENTRÄGER

Disketten

neutr. 5.25", 1D, 10er Pack	14,90
neutr. 5.25", 2D, 10er Pack	17,90
Markenfabrikate auf Anfrage	
MF 1-DD 3.5" Disk., 10er Pack	89,–

Datencassetten Industriequalität ausgesuchtes Bandmaterial

C 10 mit Box	2,20
C 10 ohne Box	2,10
C 20 mit Box	2,30
C 20 ohne Box	2,20
C 30 mit Box	2,50
C 30 ohne Box	2,40

Kunstlederhauben, beste Qualität:

464/664/6128/NLQ 401	24,90
DDI-1 / FD-1	14,90
Monitor Grün / Farbe	27,90/34,90

Lightpen	99,–
CPC 464 Grün / Farbe	auf Anfrage
CPC 6128 Grün / Farbe	998,–/1698,–
Joyce PCW 8256 / Joyce Plus	1799,–/2490,–
DMP 2000	698,–
DDI-1 / FD-1	498,–/498,–
FD-2 für Joyce	698,–
CTM 644 Farbmonitor	698,–
RS-232-Centronics-Schnittstelle Joyce	148,–
Anschlußkabel (Drucker/2. Laufwerk)	49,50
3"-Disketten Stck. / 5 Stck.	auf Anfrage
Datenrekorder / Anschlußkabel dto.	79,–/24,90
Verlängerungskabel (464) 27,90 (664/6128)	34,90
Akustikkoppler mit FTZ-Nr.	248,–
Leerhüllen für 3"-Disketten	Stck. 1,–/10 Stck. 9,–
Buchhüllen f. 3"-Disketten bzw. Cassetten	Stck. 3,90
Grafpad II Graphiktablett 464/664	275,–/295,–



Netzteil MP-2. Mit dieser Stromversorgung kann jeder CPC an ein normales Farb-TV angeschlossen werden. Besonders bei Einsatz des Rechners mit grünem Monitor stellt dies eine attraktive Erweiterung der Einsatzmöglichkeit dar.
MP-1 **128,–**
MP-2 **148,–**



3 u. 3.5" Diskettenbox mit Sortiereinrichtung und Klappklarsichtdeckel, **abschließbar** **34,90**

5.25" Diskettenbox (DX 85) für 100 Disketten mit abschließbarem Klarsichtdeckel **24,90**



RS232-Schnittstelle für den Anschluß peripherer Geräte m. serieller Schnittstelle wie Schreibmaschine, Steuergeräte, Akustikkoppler usw. Komplett mit Kabel und Stromversorgung
464/664 **148,–**
6128 **178,–**



»SFT 401« Mit dem Formulartraktor zum Schneider-Drucker »NLQ 401« wird Ihnen das umfangreiche Verarbeiten von Endlospapier wesentlich erleichtert. Die Acrylglashaube reduziert den Geräuschpegel **79,50**

Schneider Matrix Printer »NLQ 401« 50 cps, 80 Zeichen/Zeile, vorwärts- und rückwärtsdruckend, 9×9 Matrix, Korrespondenzqualität, **498,–**

Monitor-Drehfuß mit stufenloser Einstellung des Neigungswinkels für 9–14" Monitore **39,80**

R. Schuster Electronic

OBERE MÜNSTERSTR. 33 · ☎ (02305) 3770
4620 CASTROP-RAUXEL
VERSAND PER NACHNAME ZUZÜGLICH VERSANDKOSTEN.

Schneider COMPUTER DIVISION Vertragshändler
ATARI System-Fachhändler
Commodore Vertrags-Werkstatt

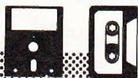


○ Senden Sie mir bitte Ihre Info zu. (2,– DM in Briefmarken liegen bei)
○ Hiermit bestelle ich per Nachnahme:

Vorname, Name
Straße, Hausnummer
PLZ, Ort
Telefonnummer
Datum, Unterschrift

Monstergarten





Schon wieder stellt sich ein mutiger Mann der Herausforderung: Die mächtige Burg "Koth" im Reiche Lythos, bekannt als Monstergarten, ist ein Ort, mit unermeßlichen Reichtümern und Schätzen.

Zählt man allerdings die Tage der Vergangenheit, so zählt man auch die tapferen Krieger und mächtigen Zauberer, die an diesem Ort des Schreckens ihr Leben gelassen haben.

Dungeons & Dragons ist ein Spielsystem aus den USA. Das Abenteuerspiel "Monstergarten" ist den D & D Fantasy-Spielen, die jetzt auch in Europa immer mehr Anhänger finden, nachempfunden.

Sie sind in den Monstergarten eingedrungen und erforschen nun dieses Labyrinth des Schreckens.
Auf dem Weg treffen Sie allerlei eigenartige Geschöpfe.

Ihre Gegner sind alle von unterschiedlicher Stärke. Die Erfahrung wird Sie lehren, wann Sie Ihnen entgegentreten können. Je mehr magische Gegenstände und Waffen Sie an sich bringen können, um so stärker werden Sie.
Diese Stärke benötigen Sie auch dringend, da der Monstergarten Sie erst wieder freiläßt, wenn Sie den "Goldenen Schlüssel der Freiheit" erbeutet haben. Selbstverständlich wird ein so wertvolles Objekt von einem besonders bösen und mächtigen Wesen bewacht.

Ein weiteres Problem ist die Versorgung mit Lebensmitteln und Proviant. Mit ein wenig kaufmännischem Geschick können Sie dieses Problem jedoch leicht bewältigen. Seien Sie jedoch nicht so raffgierig - auch im Monstergarten kennt man die Inflation.

Viel Spaß beim Durchstreifen der dunklen modrigen Gewölbe, in denen Monster, Goblins und Hexen auf Sie lauern.

(B. Morell)

```
1 ´************************************* [2798]
************************************
2 ´**                                    [104]
                                  **
3 ´**                      M O N S T E R  G [1168]
 A R T E N                        **
4 ´**                                    [104]
                                  **
5 ´**                      fuer  Scneider  [616]
 CPC  464                         **
6 ´**                       von   Bernard  [1391]
  Morell                          **
7 ´**                                    [104]
                                  **
8 ´************************************* [2798]
************************************
9 ´========================   Initialisi [6073]
erung   ============================
10 MODE 1:INK 0,0:INK 1,3:INK 2,24:INK 3,2 [3237]
6:BORDER 0
20 DIM s(40,23),t$(23),mo(28)             [899]
30 mo(1)=3:mo(2)=3:mo(3)=3:mo(4)=4:mo(5)=5 [13627]
:mo(6)=6:mo(7)=6:mo(8)=7:mo(9)=8:mo(10)=9:
mo(11)=10:mo(12)=11:mo(13)=11:mo(14)=12:mo
(15)=12:mo(16)=12:mo(17)=13:mo(18)=13:mo(1
9)=14:mo(20)=14:mo(21)=14:mo(22)=14:mo(23)
=14:mo(24)=14
40 mo(25)=15:mo(26)=15:mo(27)=16:mo(28)=17 [4504]
:zae=9:staerke=10:prov=50:taler=100
50 GOSUB 310:GOSUB 70:GOSUB 1530          [2254]
60 GOSUB 790:GOSUB 470:GOSUB 1540:GOSUB 12 [6012]
00:GOSUB 650:x=2:xn=2:y=2:yn=2:GOSUB 2680:
GOTO 1610
70 t$(1)="Boing,das war die Mauer !"      [2902]
80 t$(2)="Du bist in deiner Burg"         [1783]
90 t$(3)="Keine Angst;es ist nur ein Haend [2757]
ler"
100 t$(4)="miau,ich bin Gundels Katze"    [2586]
110 t$(5)="Ich bin die Hexe Gundel"       [2890]
120 t$(6)="Ein knochenklapperndes Skelett" [3577]
130 t$(7)="Ein schleimiger Goblin"        [3197]
140 t$(8)="Ich bin Bilbo der Gnom"        [2891]
150 t$(9)="Ich bin Frodo der Zwerg"       [2377]
160 t$(10)="Ein feuerspeiender Drache"    [3645]
170 t$(11)="Ein grausig-grusliger Geist"  [4257]
180 t$(12)="Ein schwarzer Moench des Teufe [4734]
ls"
190 t$(13)="Ich bin die gute Fee"         [1915]
200 t$(14)="Eine tollwuetige Spitzmaus"   [3001]
210 t$(15)="Ein zerflederter Zombie"      [1819]
220 t$(16)="Ein wandelnder Guhl"          [2447]
230 t$(17)="Ich,ja ich bin der Teufel pers [6938]
oenlich"
240 t$(18)="Du bist am Ziel"              [2255]
250 t$(19)="Hast du den Schluessel ? [j/n] [3868]
"
260 t$(20)="Willst Du handeln ? [j/n]"    [4036]
270 t$(21)="Kaufen oder verkaufen ? [k/v]  [3295]
Kurs: "
280 t$(22)="Du hast gewonnen"             [3222]
290 t$(23)="Du hast verloren"             [2153]
299 ´=======================   Symbole    [3622]
  ================================
300 RETURN                                [555]
310 SYMBOL AFTER 180                      [1436]
320 SYMBOL 180,0,0,6,1,205,222,126,44:´kat [2552]
ze
330 SYMBOL 181,0,2,4,2,1,93,254,62:´maus   [1975]
```

LISTINGS

```
340 SYMBOL 182,56,8,104,72,56,104,76,254:´ [2463]
hexe
350 SYMBOL 183,0,62,42,62,28,0,28,0:´skele [2768]
tt
360 SYMBOL 187,0,24,60,92,126,110,78,28:´g [3060]
nom/zwerg
370 SYMBOL 188,56,84,254,186,186,56,40,40: [3289]
´guhl/zombie
380 SYMBOL 189,20,62,106,106,126,66,86,255 [2485]
:´teufel
390 SYMBOL 184,56,124,84,84,124,70,231,255 [2380]
:´ geist
400 SYMBOL 185,96,88,28,62,42,62,99,237:´  [2850]
moench
410 SYMBOL 186,98,239,147,24,28,142,158,12 [3899]
6:´ drache
420 SYMBOL 194,255,0,0,255,16,16,16,239    [1794]
430 SYMBOL 195,187,68,68,199,4,4,4,251     [2407]
440 SYMBOL 196,247,8,8,255,225,33,33,222   [1639]
450 SYMBOL 200,108,146,146,69,161,138,100, [2437]
24
460 RETURN                                 [555]
469 ´=======================    Labyrinth  [4220]
aufbauen  =========================
470 anzahl=0                               [176]
480 FOR y=2 TO 22 STEP 2                   [1274]
490 x=INT(RND*4+2)                         [1511]
500 laenge=INT(RND*5+5)                    [1753]
510 FOR xx=x TO x+laenge                   [2001]
520 s(xx,y)=0:IF xx+1>=40 THEN 560         [1344]
530 IF y<2 OR y>22 THEN 570                [689]
540 NEXT xx                                [464]
550 x=x+2+laenge:GOTO 500                  [1997]
560 NEXT y                                 [359]
570 FOR y=3 TO 21 STEP 2                   [733]
580 FOR xx=2 TO 39 STEP INT(RND*5+2)       [2197]
590 IF s(xx,y-1)=0 AND s(xx,y+1)=0 AND s(x [3917]
x+1,y)=1 AND s(xx-1,y)=1 THEN s(xx,y)=0:an
zahl=anzahl+1
600 NEXT xx                                [464]
610 IF anzahl<7 THEN 580                   [756]
620 anzahl=0:NEXT y                        [153]
630 FOR y=2 TO 22:s(40,y)=1:NEXT y         [1781]
640 RETURN                                 [555]
649 ´=======================    Garten aus [4734]
drucken  ==========================
650 MODE 1                                 [506]
660 FOR x=1 TO 40:FOR y=1 TO 23            [1344]
670 sprung=1+s(x,y):LOCATE x,y:IF sprung>3 [2968]
 THEN sprung=4
680 ON sprung GOSUB 700,710,720,730        [1330]
690 NEXT y:NEXT x:FOR x=1 TO 40:s(x,1)=19: [5996]
s(x,23)=19:NEXT x:FOR y=2 TO 22:s(1,y)=19:
s(40,y)=19:NEXT y:RETURN
700 PAPER 0:PEN 3:PRINT " ":RETURN         [2542]
710 PAPER 0:PEN 1:PRINT CHR$(194+INT(RND*3 [4933]
)):RETURN
720 PAPER 2:PEN 1:PRINT CHR$(194+INT(RND*3 [2398]
)):RETURN
730 PAPER 0:PEN 2:PRINT CHR$(63):RETURN    [2391]
739 ´=======================    Joyabfrage [4130]
 ===================================
740 xn=x+(INKEY(74)>-1)-(INKEY(75)>-1)     [2130]
750 yn=y+(INKEY(72)>-1)-(INKEY(73)>-1)     [1989]
760 boing=0:IF JOY(0)>=16 THEN boing=1     [1404]
770 RETURN                                 [555]
779 ´=======================    Startbild  [4377]
 ==================================
780 MODE 1                                 [506]
790 PEN 3:PAPER 0:LOCATE 5,1:PRINT "M O N  [6011]
S T E R":LOCATE 22,1:PRINT "G A R T E N":P
EN 1
800 FOR y=14 TO 23:FOR x=7 TO 35           [1877]
810 LOCATE x,y:PRINT CHR$(194+INT(RND*3)): [2401]
NEXT x:NEXT y
820 FOR y=7 TO 13:FOR x=7 TO 13:LOCATE x,y [4946]
:PRINT CHR$(194+INT(RND*3)):NEXT x:NEXT y
830 FOR y=6 TO 13:FOR x=17 TO 22:LOCATE x, [5573]
y:PRINT CHR$(194+INT(RND*3)):NEXT x:NEXT y
840 FOR y=9 TO 13:FOR x=29 TO 35:LOCATE x, [4581]
y:PRINT CHR$(194+INT(RND*3)):NEXT x:NEXT y
850 PEN 1:PAPER 0:LOCATE 8,7:PRINT" ":LOCA [12432]
TE 10,7:PRINT" ":LOCATE 12,7:PRINT" ":LOCA
TE 14,14:PRINT" ":LOCATE 16,14:PRINT" ":LO
CATE 24,14:PRINT" ":LOCATE 26,14:PRINT" ":
LOCATE 28,14:PRINT" ":LOCATE 30,9:PRINT" "
:LOCATE 32,9:PRINT" ":LOCATE 34,9:PRINT" "
860 FOR y=20 TO 23:LOCATE 23,y:PRINT "     [2885]
":NEXT y
870 LOCATE 8,10:PRINT" ":LOCATE 8,11:PRINT [13147]
" ":LOCATE 10,11:PRINT" ":LOCATE 10,12:PRI
NT" ":LOCATE 12,12:PRINT" ":LOCATE 12,13:P
RINT" ":LOCATE 30,11:PRINT" ":LOCATE 30,12
:PRINT" ":LOCATE 32,12:PRINT" ":LOCATE 32,
13:PRINT" ":LOCATE 34,13:PRINT" ":LOCATE 3
4,1
880 LOCATE 18,7:PRINT" ":LOCATE 18,8:PRINT [5856]
" ":LOCATE 21,7:PRINT" ":LOCATE 21,8:PRINT
" "
890 FOR x=1 TO 4:LOCATE 7+2*x,17:PRINT" ": [5677]
LOCATE 7+2*x,18:PRINT" ":NEXT x
900 MOVE 48*2+1,152*2:FOR x=1 TO 3:GOSUB 1 [2331]
180:NEXT x:DRAWR 15,0
910 MOVE 104*2+1,88*2:DRAWR 16,0:DRAWR 0,1 [3748]
5:FOR x=1 TO 7:GOSUB 1180:NEXT x
920 MOVE 224*2+1,136*2:FOR x=1 TO 3:GOSUB  [3091]
1180:NEXT x:DRAWR 15,0
930 MOVE 144*2+1,72*2:GOSUB 1180:GOSUB 118 [2447]
0:DRAWR 15,0
940 MOVE 200*2+1,72*2:GOSUB 1180:GOSUB 118 [3490]
0:DRAWR 15,0
950 MOVE 152*2+1,16*2:DRAWR 0,80:DRAWR -16 [1998]
,0:DRAWR 0,29
960 MOVE 208*2+1,32:DRAWR 0,80:DRAWR -16,0 [2077]
:DRAWR 0,29
970 MOVE 176*2,32:DRAWR 0,80:DRAWR 16,0:DR [2868]
AWR 0,29
980 MOVE 232*2,32:DRAWR 0,80:DRAWR 16,0:DR [2173]
AWR 0,29
990 MOVE 56*2,128*2:GOSUB 1190             [590]
1000 MOVE 72*2,120*2:GOSUB 1190            [1209]
1010 MOVE 88*2,112*2:GOSUB 1190            [1434]
1020 MOVE 232*2,120*2:GOSUB 1190           [797]
1030 MOVE 248*2,112*2:GOSUB 1190           [797]
1040 MOVE 264*2,104*2:GOSUB 1190:MOVE 17*1 [3352]
6,19*16:GOSUB 1190:MOVE 20*16,19*16:GOSUB
1190
1050 FOR x=128 TO 112*2 STEP 32: MOVE x,14 [3180]
4:GOSUB 1190:NEXT x
1060 MOVE 48*2+1,32:DRAWR 128*2,0          [2041]
1070 MOVE 48*2+1,32:DRAWR 0,136*2          [2075]
1080 MOVE 26*16+1,32:DRAWR 70*2+1,0:DRAWR  [2205]
0,240
1090 MOVE 16*16+1,88*2:DRAWR 0,72*2        [1514]
1100 MOVE 22*16-1,88*2:DRAWR 0,72*2        [1324]
1110 MOVE 13*16,88*2:DRAWR 0,64*2          [1704]
1120 MOVE 28*16+1,88*2:DRAWR 0,48*2        [1094]
1130 MOVE 80*2,146*2:DRAWR 0,90            [1300]
1140 MOVE 30*16,16*16+2:DRAWR 0,90:FOR x=1 [6752]
5*16 TO 23*16 STEP 4:MOVE x,20*16+1:DRAW 1
9*16,399:NEXT x
1150 LOCATE 11,3:PAPER 3:PRINT " ";CHR$(18 [1595]
5);" "
1160 LOCATE 31,5:PAPER 3:PRINT " ";CHR$(18 [1886]
4);" "
1169 ´=======================    Lauf in   [3639]
die Burg  ========================
1170 PEN 1:PAPER 0:RETURN                  [1786]
1180 DRAWR 15,0,3:DRAWR 0,-15:DRAWR 17,0:D [2839]
RAWR 0,15:RETURN
1190 DRAWR 17,0:DRAWR 0,-29:DRAWR -17,0:DR [2734]
AWR 0,29:RETURN
1200 PEN 3:x=10:y=24                       [1434]
1210 LOCATE 10,24:PRINT CHR$(248+INT(RND*4 [2413]
))
1220 GOSUB 740:IF SQ(1)<64 THEN GOSUB 1360 [2396]
1230 IF xn=x AND yn=y AND RND<0.8 THEN 122 [4710]
0 ELSE IF xn>=23 AND xn<=26 AND yn<=24 THE
N 1250
1240 IF xn>=5 AND xn<=38 THEN yn=24:GOTO 1 [3229]
250 ELSE 1220
1250 LOCATE x,y:PRINT " ":LOCATE xn,yn:PRI [2931]
NT CHR$(248+INT(RND*4)):x=xn:y=yn
1260 IF y<=21 THEN CLS:RETURN ELSE 1220    [2295]
1269 ´=======================     Tod   == [4246]
====================================
1270 SPEED KEY 10,3:´ TOD                  [2058]
1280 CLS:PEN 0:PAPER 1:FOR i=1 TO 24:LOCAT [5187]
E 1,i:PRINT STRING$(40,CHR$(194+INT(RND*3)
)):NEXT i:PEN 3: PAPER 0
1290 FOR i=14 TO 25:FOR ii=8 TO 11:LOCATE  [5371]
i,ii:PRINT CHR$(184):NEXT ii,i
1300 FOR i=18 TO 21:FOR ii=2 TO 23:LOCATE  [7378]
i,ii:PRINT CHR$(184):NEXT ii,i:GOSUB 1310:
GOTO 1330
1309 ´=======================    Todesmel  [4259]
odie  ============================
1310 SOUND 1,284,80,15:SOUND 2,568,80,15:S [13598]
OUND 3,287,80,15:SOUND 1,190,15,15:SOUND 2
,379,15,15:SOUND 3,193,15,15:SOUND 1,179,1
```

```
5,15:SOUND 2,358,15,15:SOUND 3,182,15,15:S
OUND 1,159,15,15:SOUND 2,319,15,15:SOUND 3
,162,15,15
1320 SOUND 1,225,60,15:SOUND 2,451,60,15:S [11829]
OUND 3,228,60,15:SOUND 1,179,50,15:SOUND 2
,358,50,15:SOUND 3,182,50,15:SOUND 1,253,8
0,15:SOUND 2,506,80,15:SOUND 3,256,80,15:S
OUND 1,213,100,15:SOUND 2,426,150,15:SOUND
 3,216,100,15:RETURN
1329 '=========================  Neues Sp [5050]
iel ?  =============================
1330 LOCATE 1,24:PRINT "Neues Spiel ?  [ j [2135]
/n ]          "
1340 IF INKEY$="j" OR INKEY$="J" THEN CLEA [5220]
R:GOTO 10 ELSE IF SQ(1)<64 THEN GOSUB 1360
1350 IF INKEY$="n" OR INKEY$="N" THEN END  [1773]
ELSE 1340
1359 '=========================  Begleit [4153]
melodie  ===========================
1360 ON sp GOSUB 1390,1400,1410,1420,1430, [4230]
1440,1450,1460,1470,1480,1490,1500,1510
1370 IF sp=13 THEN sp=0                    [347]
1380 sp=sp+1:RETURN                        [1269]
1390 SOUND 1,60,40:SOUND 1,63,20:SOUND 1,6 [2883]
0,60:RETURN
1400 SOUND 1,80,30:SOUND 1,67,20:SOUND 1,7 [5558]
5,30:SOUND 1,89,100:RETURN
1410 SOUND 1,119,30:SOUND 1,89,20:SOUND 1, [2132]
80,60:RETURN
1420 SOUND 1,119,30:SOUND 1,80,20:SOUND 1, [2401]
75,60:RETURN
1430 SOUND 1,60,40:SOUND 1,63,20:SOUND 1,6 [2883]
0,60:RETURN
1440 SOUND 1,80,30:SOUND 1,67,20:SOUND 1,7 [5558]
5,30:SOUND 1,89,100:RETURN
1450 SOUND 1,119,30:SOUND 1,89,20:SOUND 1, [2132]
80,60:RETURN
1460 SOUND 1,75,20:SOUND 1,80,30:SOUND 1,8 [2521]
9,60:RETURN
1470 SOUND 1,80,40:SOUND 1,75,30:SOUND 1,6 [5361]
7,30:SOUND 1,60,40:SOUND 1,60,40:RETURN
1480 SOUND 1,56,20:SOUND 1,60,30:SOUND 1,6 [4732]
7,20:SOUND 1,134,20:SOUND 1,67,40:RETURN
1490 SOUND 1,60,20:SOUND 1,67,30:SOUND 1,7 [4840]
5,20:SOUND 1,150,20:SOUND 1,75,40:RETURN
1500 SOUND 1,67,20:SOUND 1,75,30:SOUND 1,8 [3434]
0,100:RETURN
1510 SOUND 1,60,40:SOUND 1,63,20:SOUND 1,6 [3394]
0,30:RETURN
1520 RETURN                                [555]
1529 '=====================  Spielfeldspe [4441]
icher fuellen  =====================
1530 FOR x=1 TO 40:FOR y=1 TO 23:s(x,y)=1: [4013]
NEXT y,x:RETURN
1540 FOR x=2 TO 8                          [683]
1550 FOR y=2 TO 8                          [712]
1559 '=====================  Burg + Ziel  [5539]
setzen  ============================
1560 s(x,y)=0:NEXT y,x                     [418]
1570 s(3,3)=2:s(3,4)=2:s(4,4)=2:s(4,5)=2:s [2310]
(4,6)=2:s(4,7)=2
1580 s(5,3)=2:s(5,4)=2:s(5,5)=2:s(5,6)=2:s [2187]
(5,7)=2
1590 s(6,4)=2:s(6,5)=2:s(6,6)=2:s(6,7)=2:s [2582]
(7,3)=2:s(7,4)=2
1600 x=INT(RND*10)+27:y=INT(RND*10)+13:IF  [4461]
s(x,y)=0 THEN s(x,y)=18:RETURN ELSE 1600
1609 '=====================  Hauptprogram [4378]
m  =================================
1610 GOSUB 740:CLS#1:GOSUB 2670:IF prov<=0 [4826]
 OR staerke<1 THEN GOTO 1270
1620 sprung=1+s(xn,yn):IF SQ(1)<64 THEN GO [2388]
SUB 1360
1630 ON sprung GOTO 1640,1670,1690,1730,17 [5742]
40,1770,1810,1840,1870,1900,1930,1960,2000
,2030,2040,2080,2110,2140,2170,1680
1640 LOCATE x,y:PRINT " ":LOCATE xn,yn:PRI [1865]
NT CHR$(248+INT(RND*4))
1650 IF x<>xn OR y<>yn THEN zae=zae-1:prov [7141]
=prov-1:GOSUB 2430:IF zae<0 THEN GOSUB 224
0:IF zae<-500 AND INT(RND*20)=10 THEN GOSU
B 2450
1660 x=xn:y=yn:GOTO 1610                   [2015]
1670 IF boing=1 THEN GOSUB 2480:SOUND 3,10 [4380]
00,20,15,,,2:LOCATE 1,25:PRINT t$(1):GOTO
2220
1680 xn=x:yn=y:GOTO 1610                   [2278]
1690 GOSUB 2480:LOCATE 1,25:PRINT t$(2)    [1926]
1700 GOSUB 2500:LOCATE 1,24:PRINT t$(20)   [2375]
1710 IF INKEY$="j" THEN GOSUB 2520 ELSE IF [2209]
 INKEY$="n" THEN 1720 ELSE 1710
1720 xn=x:yn=y:s(x,y)=0:GOTO 1610          [2377]
1730 GOSUB 2480:LOCATE 1,25:PRINT t$(3):GO [2126]
TO 1700
1740 GOSUB 2480:LOCATE 1,25:PRINT t$(4):st [4734]
=7:ii=180:GOSUB 2620:IF g=1 THEN GOTO 1760
1750 LOCATE 1,25:PRINT "Sie bewachte ein M [7008]
edallion":staerke=staerke+1:kt=1:GOSUB 273
0:GOTO 1610
1760 LOCATE 1,25:PRINT "Sie nimmt Dir 20 P [4763]
roviant":prov=prov-20:GOSUB 2740:GOTO 1610
1770 GOSUB 2480:LOCATE 1,25:PRINT t$(5):st [8148]
=10:ii=182:IF kt=1 THEN st=15:GOSUB 2710:G
OSUB 2480:LOCATE 1,25:PRINT "Sie ist boese
,weil ihre Katze tot ist"
1780 GOSUB 2620:IF g=1 THEN GOTO 1800      [1621]
1790 LOCATE 1,25:PRINT "Sie bewachte eine  [6015]
Allraune":staerke=staerke+2:GOSUB 2730:GOT
O 1610
1800 LOCATE 1,25:PRINT "Sie nimmt Dir 40 P [5894]
roviant":prov=prov-40:GOSUB 2740:GOTO 1610
1810 GOSUB 2480:LOCATE 1,25:PRINT t$(6):st [4922]
=15:ii=183:GOSUB 2620:IF g=1 THEN GOTO 183
0
1820 LOCATE 1,25:PRINT "Es bewachte 400 Ta [3980]
ler":taler=taler+400:GOSUB 2730:GOTO 1610
1830 LOCATE 1,25:PRINT "Es nimmt Dir 100 P [5243]
roviant":prov=prov-100:GOSUB 2740:GOTO 161
0
1840 GOSUB 2480:LOCATE 1,25:PRINT t$(7):st [5302]
=11:ii=185:GOSUB 2620:IF g=1 THEN GOTO 186
0
1850 LOCATE 1,25:PRINT "Er bewachte einen  [7991]
Zauberschild":staerke=staerke+2:GOSUB 2730
:GOTO 1610
1860 LOCATE 1,25:PRINT "Er schwaecht Dich" [6310]
:staerke=staerke-1:GOSUB 2740:GOTO 1610
1870 GOSUB 2480:LOCATE 1,25:PRINT t$(8):st [4884]
=13:ii=187:GOSUB 2620:IF g=1 THEN GOTO 189
0
1880 LOCATE 1,25:PRINT "Er bewachte einen  [6493]
Zauberhelm":staerke=staerke+2:GOSUB 2730:G
OTO 1610
1890 LOCATE 1,25:PRINT "Er schwaecht Dich" [6310]
:staerke=staerke-1:GOSUB 2740:GOTO 1610
1900 GOSUB 2480:LOCATE 1,25:PRINT t$(9):st [5100]
=10:ii=187:GOSUB 2620:IF g=1 THEN GOTO 192
0
1910 LOCATE 1,25:PRINT "Er bewachte einen  [7143]
Haufen Edelsteine":taler=taler+300:GOSUB 2
730:GOTO 1610
1920 LOCATE 1,25:PRINT "Er nimmt dein ganz [4936]
es Geld":taler=0:GOSUB 2740:GOTO 1610
1930 GOSUB 2480:LOCATE 1,25:PRINT t$(10):s [4524]
t=17:ii=186:GOSUB 2620:IF g=1 THEN GOTO 19
50
1940 LOCATE 1,25:PRINT "Er bewachte eine g [8269]
oldene Ruestung":staerke=staerke+3:GOSUB 2
730:GOTO 1610
1950 LOCATE 1,25:PRINT "Er nimmt Dir ...": [9334]
taler=taler-200:prov=prov-100:staerke=stae
rke-INT(RND*2):GOSUB 2740:GOTO 1610
1960 GOSUB 2480:LOCATE 1,25:PRINT t$(11):s [4288]
t=10:ii=184:GOSUB 2620:IF g=1 THEN GOTO 19
90
1970 LOCATE 1,25:PRINT "Er bewachte ...";: [4128]
IF RND<0.3 THEN PRINT "Ring":staerke=staer
ke+1
1980 prov=prov+50:GOSUB 2730:GOTO 1610     [2683]
1990 LOCATE 1,25:PRINT "Er nimmt Dir 60 Pr [5275]
oviant":prov=prov-60:GOSUB 2740:GOTO 1610
2000 GOSUB 2480:LOCATE 1,25:PRINT t$(12):s [3684]
t=9:ii=185:GOSUB 2620:IF g=1 THEN GOTO 202
0
2010 LOCATE 1,25:PRINT "Er bewachte 30 Pro [5718]
viant":prov=prov+30:GOSUB 2730:GOTO 1610
2020 LOCATE 1,25:PRINT "Er nimmt Dir 40 Pr [5829]
oviant":prov=prov-40:GOSUB 2740:GOTO 1610
2030 GOSUB 2480:LOCATE 1,25:PRINT t$(13):G [13920]
OSUB 2500:LOCATE 1,24:PRINT"Sie spendet Mu
t und Kraft !!!":staerke=staerke+INT(RND*3
+1):prov=prov+50:s(xn,yn)=0:LOCATE xn,yn:P
RINT " ":xn=x:yn=y:GOSUB 2710:GOTO 1610
2040 GOSUB 2480:LOCATE 1,25:PRINT t$(14):s [5790]
t=6:ii=181:GOSUB 2620:IF g=1 THEN GOTO 207
0
2050 LOCATE 1,25:PRINT "Sie bewachte 10 Pr [8232]
oviant";:IF RND<0.1 THEN PRINT" und einen
Ring":staerke=staerke+1
2060 prov=prov+10:GOSUB 2730:GOTO 1610     [2657]
```

```
2070 LOCATE 1,25:PRINT "Sie nimmt Dir 15 P [5243]
roviant":prov=prov-15:GOSUB 2740:GOTO 1610
2080 GOSUB 2480:LOCATE 1,25:PRINT t$(15):s [3735]
t=14:ii=188:GOSUB 2620:IF g=1 THEN GOTO 21
00
2090 LOCATE 1,25:PRINT "Er bewachte ein Za [5594]
uberschwert":staerke=staerke+2:GOSUB 2730:
GOTO 1610
2100 LOCATE 1,25:PRINT "Er nimmt Dir 80 Pr [5062]
oviant":prov=prov-80:GOSUB 2740:GOTO 1610
2110 GOSUB 2480:LOCATE 1,25:PRINT t$(16):s [3887]
t=16:ii=188:GOSUB 2620:IF g=1 THEN GOTO 21
30
2120 LOCATE 1,25:PRINT "Er bewachte eine Z [5898]
auberaxt":staerke=staerke+2:GOSUB 2730:GOT
O 1610
2130 LOCATE 1,25:PRINT "Er schwaecht Dich" [5520]
:staerke=staerke-2:GOSUB 2740:GOTO 1610
2140 GOSUB 2480:LOCATE 1,25:PRINT t$(17):s [5468]
t=22:ii=189:GOSUB 2620:IF g=1 THEN GOTO 21
60
2150 LOCATE 1,25:PRINT "Er bewachte den Sc [6315]
hluessel":schluessel=1:GOSUB 2730:GOTO 161
0
2160 LOCATE 1,25:PRINT "Er schwaecht Dich" [8837]
:staerke=staerke-INT(RND*3+1):prov=prov-IN
T(RND*3+1)*50:taler=0:GOTO 1610
2170 GOSUB 2480:LOCATE 1,25:PRINT t$(18):G [3271]
OSUB 2500:LOCATE 1,24:PRINT t$(19)
2180 IF INKEY$="j" THEN 2190 ELSE IF INKEY [2356]
$="n" THEN 2210 ELSE 2180
2190 IF schluessel=1 THEN GOTO 2750 [2651]
2200 GOSUB 2500:LOCATE 1,24:PRINT "Du bist [7789]
 ein altes Luegenmaul":xn=x:yn=y:GOSUB 271
0:GOTO 1610
2210 GOSUB 2500:LOCATE 1,24:PRINT "Dann mu [8263]
sst Du ihn erst suchen !!!":xn=x:yn=y:GOSU
B 2710:GOTO 1610
2219 '==================== Mauer durchb [3923]
rechen =============================
2220 staerke=staerke-1:LOCATE xn,yn:PRINT [6774]
CHR$(200):FOR i=1 TO 1000 STEP 50:SOUND 3,
1000+i,1,15,,,30:NEXT i
2230 s(xn,yn)=0:GOTO 1640 [1068]
2239 '==================== Monster ersc [3793]
heint ===============================
2240 ma=ma+1:IF ma>28 THEN ma=29:RETURN [3049]
2250 PEN 2:u=INT(RND*28+1):IF mo(u)=0 THEN [2281]
 2250
2260 LOCATE x,y:ON u GOSUB 2280,2280,2280, [8367]
2290,2300,2310,2310,2320,2330,2340,2350,23
60,2360,2370,2370,2370,2380,2380,2390,2390
,2390,2390,2390,2390,2400,2400,2410,2420
2270 PEN 3:s(x,y)=mo(u):mo(u)=0:zae=INT(RN [3426]
D*15)+10:RETURN
2280 LOCATE x,y:PRINT CHR$(248):RETURN [1575]
2290 LOCATE x,y:PRINT CHR$(180):RETURN [1845]
2300 LOCATE x,y:PRINT CHR$(182):RETURN [1375]
2310 LOCATE x,y:PRINT CHR$(183):RETURN [2594]
2320 LOCATE x,y:PRINT CHR$(185):RETURN [1241]
2330 LOCATE x,y:PRINT CHR$(187):RETURN [1863]
2340 LOCATE x,y:PRINT CHR$(187):RETURN [1863]
2350 LOCATE x,y:PRINT CHR$(186):RETURN [2106]
2360 LOCATE x,y:PRINT CHR$(184):RETURN [1575]
2370 LOCATE x,y:PRINT CHR$(185):RETURN [1241]
2380 LOCATE x,y:PRINT CHR$(42):RETURN [1827]
2390 LOCATE x,y:PRINT CHR$(181):RETURN [1820]
2400 LOCATE x,y:PRINT CHR$(188):RETURN [1608]
2410 LOCATE x,y:PRINT CHR$(188):RETURN [1608]
2420 LOCATE x,y:PRINT CHR$(189):RETURN [1313]
2429 '===================== Schwierigke [4933]
itskontrolle =====================
2430 IF taler>5000 THEN CLS#1:LOCATE 1,24: [8232]
PRINT"Monsterinflation":LOCATE 1,25:PRINT"
Du erhaeltst 50 Taler Startkapital":GOSUB
1310:taler=50:RETURN
2440 IF staerke>20 AND INT(RND*100)=50 THE [2561]
N 2450 ELSE 2460
2450 INK 3,0,26:INK 0,26,0:SPEED INK 5,2:G [11212]
OSUB 2710:INK 3,26:INK 0,0:CLS#1:LOCATE 1,
25:PRINT"Ein gewaltiges Monstergewitter !?
":GOSUB 2710:staerke=staerke-INT(RND*3+2)
2460 IF staerke>20 AND INT(RND*30)=1 THEN [12062]
LOCATE xn,yn:GOSUB 710:PEN 3:CLS#1:LOCATE
1,24:PRINT"Das war der schlaue,graue Mauer
bauer !?":s(xn,yn)=1:xn=x:yn=y:GOSUB 2710:
RETURN
2470 RETURN [555]
2479 '===================== Textfenster [4784]
 loeschen =======================
2480 LOCATE 1,25:PRINT " [2198]
                     "
2490 RETURN [555]
2500 LOCATE 1,24:PRINT " [2070]
                     "
2510 RETURN [555]
2519 '===================== Handel trei [4211]
ben =================================
2520 k=INT(RND*7+3):GOSUB 2500:LOCATE 1,24 [2743]
:PRINT t$(21);k
2530 IF INKEY$="k" THEN 2540 ELSE IF INKEY [2936]
$="v" THEN 2580 ELSE 2530
2540 r=1:GOSUB 2480:LOCATE 1,25:INPUT"Wiev [3278]
iel Proviant";m$:GOTO 2690
2550 m=VAL(m$):IF taler<m*k THEN GOSUB 266 [3164]
0:GOTO 2570
2560 taler=taler-m*k:prov=prov+m:GOSUB 267 [2635]
0:IF s(xn,yn)=3 THEN 2720
2570 RETURN [555]
2580 r=2:GOSUB 2480:LOCATE 1,25:INPUT"Wiev [4912]
iel Proviant";m$:GOTO 2690
2590 m=VAL(m$):IF m>prov THEN GOSUB 2660:G [2785]
OTO 2610
2600 prov=prov-m:taler=taler+m*k:GOSUB 267 [4814]
0:IF s(xn,yn)=3 THEN 2720
2610 RETURN [555]
2619 '====================== Kampf mit d [3544]
em Monster ========================
2620 FOR i=1 TO 10:LOCATE x,y:PRINT CHR$(2 [10551]
00):SOUND 3,1000,20,15,,,1:LOCATE x,y:PRIN
T CHR$(248):LOCATE xn,yn:PRINT CHR$(200):S
OUND 1,2000,20,15,,,1:LOCATE xn,yn:PRINT C
HR$(ii):NEXT i
2630 CLS#1:IF st+INT(RND*5)>staerke+INT(RN [5156]
D*5) THEN g=1:GOSUB 2500:LOCATE 1,24:PRINT
 t$(23) ELSE g=0:GOSUB 2500:LOCATE 1,24:PR
INT t$(22)
2640 RETURN [555]
2649 '====================== Falsche Ein [4399]
gaben ============================
2650 GOSUB 2480:LOCATE 1,25:PRINT"Na,Na ni [4738]
cht schummeln !":GOSUB 2710:RETURN
2660 GOSUB 2480:LOCATE 1,25:PRINT"Kannst D [5666]
u nicht rechnen ???":GOSUB 2710:RETURN
2669 '====================== Werte ausge [3620]
ben ================================
2670 PEN 2:PAPER 1:LOCATE 9,1:PRINT USING [5407]
"######";prov:LOCATE 23,1:PRINT USING "###
###";taler:LOCATE 10,23:PRINT USING "###";
staerke:PEN 3:PAPER 0:RETURN
2679 '====================== Beschriftun [5563]
g ==================================
2680 PEN 2:PAPER 1:LOCATE 1,1:PRINT "Provi [8367]
ant:":LOCATE 16,1:PRINT "Taler:":LOCATE 1,
23:PRINT "Staerke:":PEN 3:PAPER 0:WINDOW #
1,1,40,24,25:RETURN
2689 '====================== Eingabe pru [3694]
efen ===============================
2690 FOR i=1 TO LEN(m$):a$=MID$(m$,i,1):IF [4513]
 ASC(a$)<48 OR ASC(a$)>57 THEN GOSUB 2650:
NEXT i:IF r=1 THEN 2540 ELSE 2580
2700 IF r=1 THEN 2550 ELSE 2590 [1974]
2709 '====================== Warteschlei [4977]
fe =================================
2710 FOR warten=1 TO 2000:NEXT warten:RETU [2506]
RN
2719 '====================== Monster bes [4878]
eitigen ? =========================
2720 LOCATE xn,yn:PRINT" ":s(xn,yn)=0:RETU [2574]
RN
2730 FOR i=1 TO 1000 STEP 10:SOUND 1,1020- [4225]
i,1,10:NEXT i:LOCATE xn,yn:PRINT " ":s(xn,
yn)=0
2740 xn=x:yn=y:GOSUB 2710:RETURN [2584]
2749 '====================== Ende des Ab [4321]
enteuers =========================
2750 GOSUB 780 [921]
2760 PEN 3:LOCATE 1,24:PRINT"Bravo,Du bist [4015]
 ein ein guter Abenteurer":GOSUB 1310
2770 GOSUB 2500:GOTO 1330 [1212]
2780 '*********************************** [2519]
*************************************
2781 '**      HIMEM-FRE("")=23012    / [im S [3680]
tartzustand] =17700               **
2782 '**      REM-Zeilen haben keinen Einlu [3258]
ss auf das Programm               **
2783 '**      Dateiname : MONSTERG.BAS      [4210]
      Start mit RUN                 **
2784 '*********************************** [2519]
*************************************
```

Schnittstelle RS-232 im Selbstbau

Diese Bauanleitung ist ein wahrer Leckerbissen für alle DFÜ-Fans und andere Freunde der seriellen Datenübertragung. Der niedrige Preis und die universelle Einsetzbarkeit sollten Grund genug für die fortgeschrittenen Bastelfreaks sein, unverzüglich zum Lötkolben zu greifen.

Wie Sie aus der Schaltungsbeschreibung entnehmen können, ist dieser Beitrag für Hardware-Einsteiger wohl weniger geeignet; für diese wird jedoch in unserer Serie "SCHNEIDERWARE" eine ähnliche serielle Schnittstelle mit Grundlagenwissen von der Pike auf vorgestellt werden.

Der Selbstkostenpreis dieser seriellen Schnittstelle beläuft sich je nach Angebot auf ca. 50,– bis 70,– DM. Es wurde ein integrierter V-24-Treiber mit internen Spannungswandlern eingesetzt, wodurch kein zusätzliches Netzteil benötigt wird. Baudrate und Übertragungsformat werden mit INP- und OUT- Befehlen per Software festgelegt. Eine Anpassung an verschiedene DFÜ-Programme ist somit unproblematisch.

Die Schnittstelle kann im Bereich &F0DC - &FFDC adressiert werden, wobei für den CPC die Adressen &F8DC, &F9DC, &FADC und &FBDC vorgesehen sind (siehe auch CPC-Handbuch).

Kernteil der Schaltung ist der "ACIA" MC6850, ein ASYNCHRONER COMMUNICATIONS INTERFACE ADAPTER aus der 6800-uP-Familie, der wegen seines relativ niedrigen Preises von ca. 5,– DM auch für Z-80-Systeme von Interesse ist. Der Schaltkreis besteht aus einem seriellen Sender und Empfänger, je einem Datenregister, einem Statusregister und einem Steuerregister mit einer entsprechenden Steuerlogik.

Schaltungsbeschreibung

STEUERREGISTER

Mit dem im Steuerregister abgelegten 8-Bit-Wort "wxxyyyzz" werden alle notwendigen Betriebsparameter eingestellt.

RESET und **BAUDRATE** "zz":

Bit 0 und Bit 1 legen die Baudrate fest bzw. führen einen Reset durch. Die Baudrate wird gebildet, indem die Taktfrequenz intern durch 64, 16 oder 1 geteilt wird. Bei 19.2 kHz (38.4 kHz), wie in der vorliegenden Schaltung, sind folgende Baudraten möglich:

wxxyyy10	**->Clk /64**	**300 Bd**
	int.Syn.	**(600 Bd)**
wxxyyy01	**->Clk /16**	**1200 Bd**
	int.Syn.	**(2400 Bd)**
wxxyyy00-	**>Clk /1**	**19200 Bd**
	ext.Syn.	**(38400 Bd)**
wxxyyy11-	**>RESET**	

ÜBERTRAGUNGSFORMAT "yyy":

Bit 2, 3 und 4 bestimmen das Übertragungsformat. Insgesamt können 8 verschiedene Ü-Formate vereinbart werden:

1. wxx000zz->7-Bit, 2 Stopbits gerade Parität
2. wxx001zz->7-Bit, 2 Stopbits ungerade Parität
3. wxx010zz->7-Bit, 1 Stopbit gerade Parität
4. wxx011zz->7-Bit, 1 Stopbit ungerade Parität
5. wxx100zz->8-Bit, 2 Stopbits
6. wxx101zz->8-Bit, 1 Stopbit
7. wxx110zz->8-Bit,,1 Stopbit gerade Parität
8. wxx111zz->8-Bit, 1 Stopbit ungerade Parität

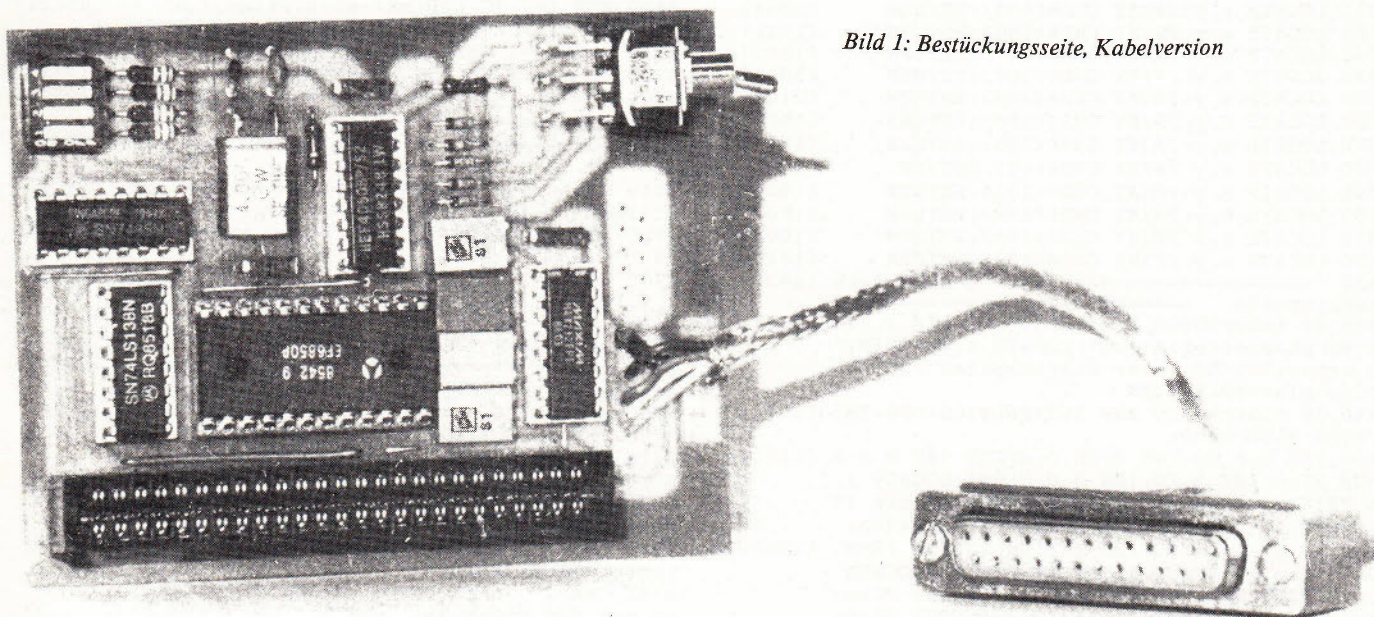


Bild 1: Bestückungsseite, Kabelversion

Telefonische Bestellung

Tel. 07131 52065

vortex
VERSANDSERVICE

Das ist keine Vortex-Werbung, sondern unser neuer Vortex-Versand-Service.
Kompetente Hard- und Software sowie Zubehör aus einer Hand!
Alle Produkte sind von uns getestet und entsprechen unserem hohen Qualitätsstandard.

JOYSTICKS:

Competition Pro 5000	54,--DM
Quickshot I	17,90DM
Quickshot II	21,90DM
Qickshot III (mit 3 Griffen)	27,90DM
Quickshot VII (Joycard Controller)	24,90DM.
Wico:The Boss	49,90DM
Schneider JY-2	39,50DM

GRAPHISCHE HILFSMITTEL:

Lightpen		99,--DM
Graffpad II	für 464 u.664:	278,--DM
Graffpad II	für 6128:	298,--DM
AMX-Mouse		278,--DM
Graphikmaster Disk: 5,25"/3"/3,5":		79.-/89,-/89,--DM

VERBINDUNGSKABEL:

Druckerkabel:	
für CPC 464,664 (1m Länge)	39,--DM
für CPC 464,664 (2m Länge)	44,--DM
für CPC 6128 (1m Länge)	39,--DM
für CPC 6128 (2m Länge)	44,--DM
für CPC 6128 (1,5m Länge)	49,--DM
Akkustikkopplerkabel(zw.RS 232 u.Modem)1,5m:	49,50DM
Anschlußkabel: 2.Floppy an CPC 664:	44,50DM
Anschlußkabel: 2.Floppy an CPC 6128:	44,50DM
Floppyverlängerungskabel für alle CPCs (2m):	59,00DM
Monitorverlängerung für CPC 464:	22,90DM
Monitorverlängerung für CPC 664 u.6128	28,90DM
Joystickverlängerung für 1 Joystick (2m Länge)	14,90DM
Recorderanschluss(CPC an 5-pol.DIN Buchse)	17,90DM
Recorderanschluss(CPC an Klinkenbuchse)	17,90DM
CPC-Stereokabel zum Anschluss an HiFi-Anlage	15,90DM

DISKETTEN:

5,25"	VORTEX-DISK DS/DD 96 tpi	10Stk.	59,--DM
3,5"	VORTEX-DISK DS/DD 135 tpi	10Stk.	89,--DM
3"	DISK CF-2	5Stk./10Stk.	52,--DM/99,--DM
3"	DISK CF-2DD (für Joyce 8512)	5Stk.	99,--DM

NÜTZLICHES ZUBEHÖR:

Plastikschablone CPC 464:Legen Sie die Schablone auf Ihre Tastatur und die ewige Blätterei im Handbuch entfällt.Folgende Anweisungen befinden sich auf der Schablone:Basic Anweisungen,Ein-/Ausgabeanweisungen, logische Operationen, mathematische Operationen,Syntax Notizen,Basic Befehle,Basic Funktionen,Numerische Funktionen,Zeichenketten,Farbtabelle und AMSDOS-Befehle
Preis: 27,90DM

Kabelschellen:Schluss mit dem Kabelsalat unter Ihrem Computertisch.Benutzen Sie unsere selbstklebende Kabel schellen. Kein Beschädigen der Tischoberfläche.In den Farben weiss,schwarz und braun lieferbar .
Verpackungsinhalt:16 Stück 8,90DM

VORTEX-Monitorständer: Dreh-und schwenkbar in allen Richtungen. Für alle 12" Monitor.Solide Ausführung aus bruchfestem Kunststoff. Preis: 49,90DM

DFÜ(Datenfernübertragung)

VORTEX-VAK-300:
Übertragungsgeschwindigkeit:300 Baud
Orginate-/Answermodus
Stromversorgung:9V Blockbatterie/externes Netzteil
Preis: 198,--DM

VORTEX-RS-232 Schnittstelle:
Ins ROM eingebundene Software mit RSX-Befehlen unter anderem:Einstellung der Parameter,übertragung(senden u. empfangen)von:einzelner Zeichen und Zeichenketten (Strings),ASCII-Files mit und ohne Softwareprotokoll, serielle Druckeransteuerung(Plotter)und integriertes Terminalprogramm.Unter CP/M:sofortige Einbindung mittels der im AMSDOS und VDOS 2.1 enthaltenen Treiberroutinen.Ausführlichste deutsche Dokumentation (ca.60 Seiten).Zeigen Sie uns eine RS-232 Schnittstelle die bei diesem Preis mehr bietet: 298,--DM
VORTEX-Phono-Set: bestehend aus:
Akkustikkopler VORTEX-VAK-300,Schnittstelle VORTEX-RS-232,Netzteil zur Stromversorgung,Diskettensoftware und Verbindungskabel.Ihr Vorteil:Alles aus einer Hand,d.h. keine Kompatibilitätsprobleme.Nur auspacken anschliessen und "datenfernübertragen".
SONDERPREIS: 498,--DM

Monitorentspiegelungsspray:Beseitigen Sie lästige Reflexionen sofort und dauerhaft.Die Augen werden spürbar entlastet.Auch für andere Anwendungen geeignet wie z.B.:Bilderglas,Autoarmaturen, Fernsehgeräte und ähnliche optische Anwendungen.Preis: 29,90DM

Datenrecorder incl.Anschlusskabel:Sonderpreis 99,--DM

BÜCHER:

Suchen Sie spezielle Computerbücher,rufen Sie uns an, wir haben ein reichhaltiges Angebot ständig auf Lager.

DISKETTENBOXEN:

3" Diskbox für 10 Disketten		14,80DM
3" Diskbox für 40 Disketten		38,90DM
dto.	abschliessbar	53,90DM
5,25" Diskbox für 50 Disketten	abschliessbar	36,90DM
5,25" Diskbox für 85 Disketten	abschliessbar	39,90DM
3,5" Diskbox für 40 Disketten		38,90DM
3,5" Diskbox für 80 Disketten		44,90DM

PFLEGEMITTEL:

ORGINAL VORTEX-ABDECKHAUBEN:
Schützen Sie nicht nur Ihre Schneider-,sondern auch Ihre VORTEX-Hardware vor Verschmutzung:

Schneider Floppy DDI-1	16,80DM
VORTEX Floppy F1-S o.F1-D	19,80DM
Schneider Konsole für 464 u.664	19,80DM
Schneider Konsole für 6128	19,80DM
VORTEX Floppy F1-X u. M1-X	19,80DM
Schneider Monitor grün	24,80DM
Schneider Monitor color	26,80DM
Schneider NLQ 401	19,80DM
Schneider DMP 2000	22,80DM
VORTEX Winchesterlaufwerk WD-20	19,80DM

Reinigen Sie Ihr 5,25"Laufwerk mit unserem speziellem Reinigungsset:
5,25"Kopfreinigungsdiskette: 13,90DM

Kassettenreinigungsset bestehend aus:
1 Prüfkassette,1 Reinigungsmittel für Kapstan-u.Gummiantrieb,1 Tonkopfreinigungsflüssigkeit und mehrere Reinigungsstäbchen.Einführungspreis: 10,90DM

DRUCKER:

Typenraddrucker VORTEX-2000	998,--DM
Matrix-VORTEX 85(baugleich Epson FX-85):	1199,--DM
Star NL-10	898,--DM
Panasonic KX-P 1080	798,--DM
Panasonic KX-P 1090	698,--DM
Panasonic KX-P 1092	1098,--DM
Citizen LSP-10	948,--DM
Schneider DMP 2000	678,--DM

Farbbänder für:	
NLQ 401,MPS 803,Brother 1009,Centronics GLP:	13,90DM
Panasonic KX-P 1080,1090,1091,1092,1592:	29,90DM
Riteman F+/C+,DMP 2000	28,90DM
Epson RX 80,FX 80,FX 85,MX 70,MX 80, MX 82,IBM 5152:	14,80DM
Joyce-Drucker,Seikosha SP 800, SP 1000,Thomson PR90:	24,80DM
Seikosha GP 500,GP 550	11,90DM
Okidata ML 182,ML 183,ML 192,ML 193	19,90DM

Druckerpapier:	
300 Blatt EDV-Endlospapier im Tragekarton	14,90DM
1.000 Blatt 240mm x 12" endlos,holzfrei 60g	29,90DM

Druckerständer:
Papierzuführung wahlweise von unten oder hinten,sehr stabile Ausführung,Nutzfläche: 40 x 32 cm 55,--DM

Bei Schneider-Produkten aktuelle Tagespreise bitte telefonisch erfragen.

JOYCE--HARDWARE:

Schneider PCW 8256 Joyce	1.648,--DM
Schneider PCW 8512 Joyce plus	2.298,--DM
Aufrüstkit für Joyce PCW 8256 auf 512kB	148,--DM
FD-2(1 MB-Laufwerk als 2.Floppy)	698,--DM
Bildschirmfilter für Joyce-Monitor	89,--DM
Joyce-Drucker Verlängerungskabel	59;--DM

JOYCE--SOFTWARE:

Locoscript(Kurzanleitung mit Übungsdisc)	29,50DM
DR-Graph	199,50DM
DR-Draw	199,50DM
Prompt(Dateiverwaltung m. Listengenerator)	69,--DM
Geschäftslösung:	
Adress-,Lager-und Fakturierungsprgr.	199,50DM
Wordstar,d BaseII und Multiplan je	199,--DM
Quick-Name(Adressverwaltung für3.400 Personen)	128,--DM

Weitere Informationen über diese Programme bitte anfordern!

NEU-NEU-NEU-NEU-NEU-NEU-NEU-NEU-NEU-NEU-NEU-NEU-NEU-NEU-
SOFTWARE-SOFTWARE-SOFTWARE-SOFTWARE-SOFTWARE-SOFTWARE-

PAPERBOY	C/D	36,--/54,--DM
1942	C/D	36,--/54,--DM
LIGHT FORCE	C/D	36,--/54,--DM
BOMB JACK	C/D	32,--/45,--DM
GHOST'N GOBLINS	C/D	29,90/39,90DM
LEADER BOARD	C/D	39,--/49,--DM
GREEN BERET	C/D	35,--/45,--DM
BAT MAN	C/D	35,--/45,--DM
EQUINOX	C/D	35,--/45,--DM
KNIGHT GAMES	C/D	39,--/59,--DM
HEXENKÜCHE TEIL II	C/D	29,--/45,--DM
IMPOSSIBLE MISSION	C/D	39,--/59,--DM
SHOGUN	C/D	39,--/59,--DM
MISSION ELEVATOR	C/D	39,--/59,--DM
BEACH HEAD TEIL II	C/D	39,--/59,--DM
GOONIES	C/D	39,--/59,--DM
REBEL PLANET	C/D	39,--/59,--DM
QUESTPROBE	C/D	39,--/59,--DM
CONTERMINATION	C/D	39,--/56,--DM
INTERNATIONAL KARATE	C/D	29,90/39,90DM
STREET HAWK	C/D	39,--/59,--DM

Selbstverständlich erhalten Sie über den Vortex-Versand-Service auch alle Vortex-Produkte!

Händleranfragen erwünscht!
Achtung Programmierer: Haben Sie gute Spiele (keine Adventures) oder Utilities, setzen Sie sich mit uns in Verbindung. Wir sind ein fairer Partner.

Telefonische Bestellung von 8 - 12 Uhr und von 13 - 17 Uhr.
Außerhalb der Geschäftszeiten nimmt unser Anrufbeantworter Ihre Bestellung entgegen.

vortex Computersysteme GmbH
Falterstrasse 51 - 53
7101 Flein
Tel. 07131/52065
Tx 728915 vortx d

O Senden Sie mir Ihren Katalog (Schutzgebühr DM 3.-)
O Senden Sie mir umgehend folgende Artikel aus Ihrem Angebot:
O per Nachnahme O per Euro-Scheck

_____ DM
_____ DM
_____ DM
_____ DM
bei Aufträgen bis DM 200.- Versandkostenpauschale DM 5.90 _____ DM
Gesamtsumme _____ DM

Absender:

Tel. Nr.

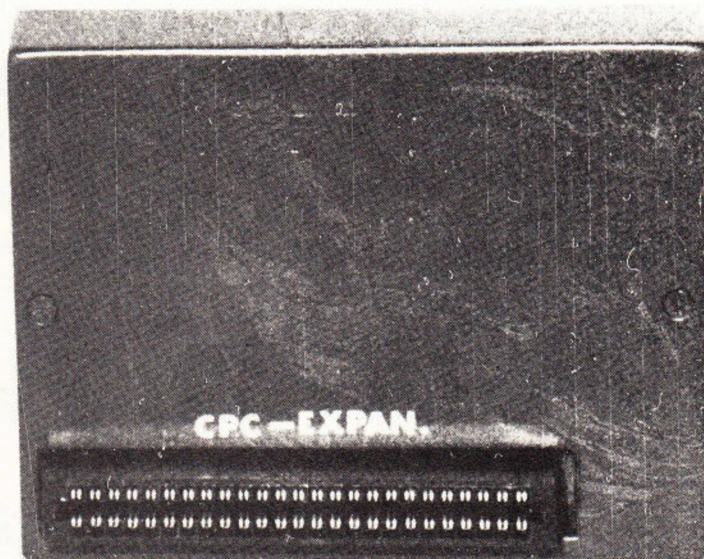
Unterschrift:

Alle Lieferungen erfolgen auf Grund unserer Allgemeinen Geschäftsbedingungen

INTERRUPTSTEUERUNG "xx": Senderseitig

Mit Bits 5 und 6 wird festgelegt, wie sich die Statusregistermeldung "SENDERDATENREGISTER leer" auswirken soll. Es sind 4 verschiedene Kombinationen möglich:

1. w00yyyzz->$\overline{\text{RTS}}$ = L
kein Interrupt
2. w01yyyzz >$\overline{\text{RTS}}$ = L
Interrupt
3. w10yyyzz->$\overline{\text{RTS}}$ = H
kein Interrupt
4. w11yyyzz->$\overline{\text{RTS}}$ = L
kein Interrupt
Leerzeichenausgabe

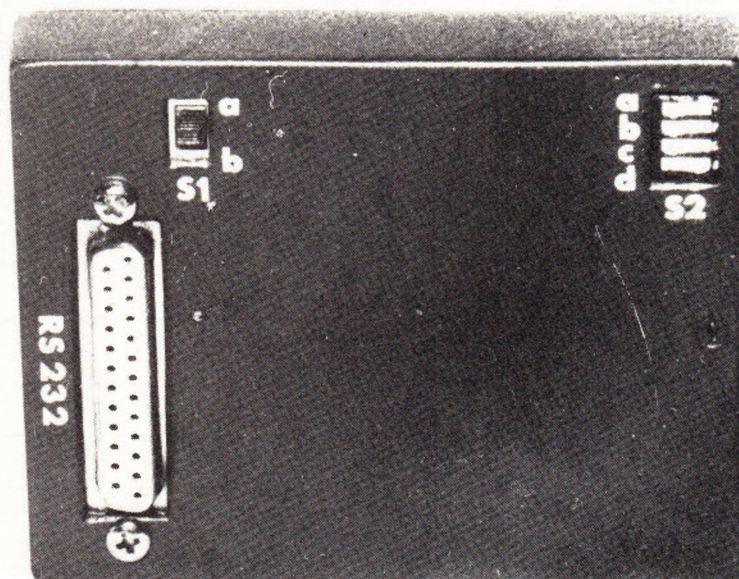


Auch im Gehäuse ...

Der Ausgang $\overline{\text{IRQ}}$ blieb in der Schaltung unberücksichtigt, kann jedoch über die Abfrage des Statusregisters softwaremäßig behandelt werden.

INTERRUPTSTEUERUNG: Empfängerseitig "w"
1xxyyyzz
Wird dieses Bit gesetzt, erfolgt ein Interrupt bei folgenden Statusmeldungen:

1. EMPFÄNGERDATENREGISTER voll
2. EMPFÄNGERDATENREGISTER Überlauf
3. $\overline{\text{DCD}}$-Eingang L->H (Data-Carrier-Detect)



... ist die Schnittstelle sehr kompakt

In der vorliegenden Schaltung liegt $\overline{\text{DCD}}$ fest auf L-Pegel.

Statusregister

Das Statusregister kann nur gelesen werden. Dieses geschieht im einfachsten Fall durch den Basic-Befehl:

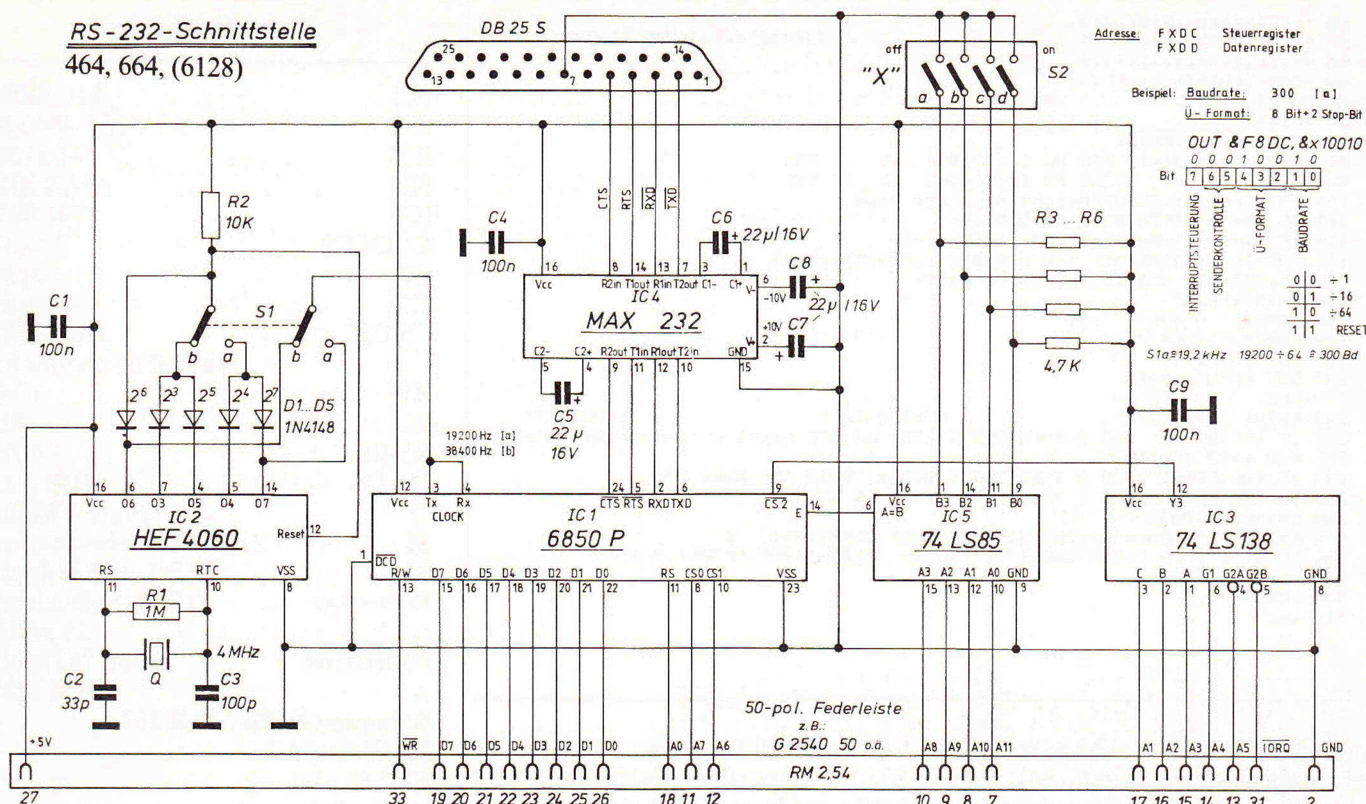
"PRINT INP (Adr.)". Man erhält eine Zahl, die erst nach einer Umwandlung in ein 8-Bit-Binärformat eine übersichtliche Aussage über die Betriebszustände des ACIA zuläßt.

Bit 0 ->1:
Datenregister des Empfängers ist voll. Das übertragene Wort ist komplett und kann im Rechner weiterverarbeitet werden.
Bit 1 ->1:
Datenregister des Senders ist leer. Es kann ein neues Zeichen gesendet werden.
Bit 2 ->1:
Ein Modem hat signalisiert, daß kein Datenträger auf der Ü-Strecke vorhanden ist. In der vorliegenden Schaltung ist jedoch Bit 2 immer auf 0.
Bit 3: Es wird der logische Zustand des $\overline{\text{CTS}}$-Einganges angezeigt.
Bit 3 ->0:
Modem signalisiert: "CLEAR TO SEND"
Bit 3 ->1:
Sender stoppt die Übertragung von Daten
Bit 4 ->1:
Synchronisation oder Datenübertragung der Ü-Strecke ist nicht in Ordnung.
Bit 5 ->1:
Datenüberlauf auf der Empfangsstrecke. Daten wurden zwar empfangen, aber nicht im Datenregister abgespeichert.
Bit 6 ->1:
Es liegt ein Paritätsfehler vor. Die Parität der empfangenen Bits entspricht nicht der vereinbarten Parität.
Bit 7 ->1:
Der Interruptausgang ist aktiviert (L-Pegel). Dieses Bit wird zurückgesetzt entweder durch das Schreiben des Senderdatenregisters oder durch das Lesen des Empfangsregisters.

Programmierung des ACIA

Beispiel:

Adresse	F8DCh	Steueregister
Adresse	F8DDh	Datenregister



Schaltplan: die komplette Schaltung. Per Adapterkabel ist natürlich auch ein Anschluß an den 6128 möglich.

Einfachste Befehlssequenz:

1. OUT &F8DC,3
   Masterreset

2. Out &F8DC,20+1
   Ü-Format vereinbaren:
   kein Interrupt
   8-Bits
   1 Stopbit
   1200 Baud

3. IF BIN$(INP(&F8DC))
   AND 2
   THEN OUT &F8DD,'Datum'
   Abfrage des Statusregisters, ob Senderegister leer, wenn ja, dann Zeichen aussenden.

4. IF BIN$(INP(&F8DC))
   AND 1
   THEN 'Datum'=INP(&F8DD)
   Abfrage des Statusregisters, ob Empfangsregister voll, wenn ja, dann Zeichen einlesen.

Aufbau und Inbetriebnahme

Die Platine wird bis auf die IC's nach Bestückungsplan bestückt. Die 25-polige D-Buchse und der 2-polige Schiebeschalter werden von der Lötseite aufgelötet (siehe Bestückungsplan). Um die mechanische Belastung des Erweiterungsports möglichst gering zu halten, empfiehlt es sich, die D-Buchse mit einem Kabelschwanz anzuschließen (Foto). Steht keine Federleiste (Slot) in Wire-Wrap-Ausführung mit langen Anschlußstiften zur Verfügung, sollte man auf eine minimale Bauteilhöhe achten, damit beim Aufstecken der Schaltung auf den Erweiterungsbus des CPC keine Platzprobleme auftreten. Zwangsläufig muß dann auch auf IC-Sockel verzichtet werden. Die IC's sollten in folgender Reihenfolge eingelötet und die einzelnen Schaltungsteile separat auf ihre Funktion überprüft werden:

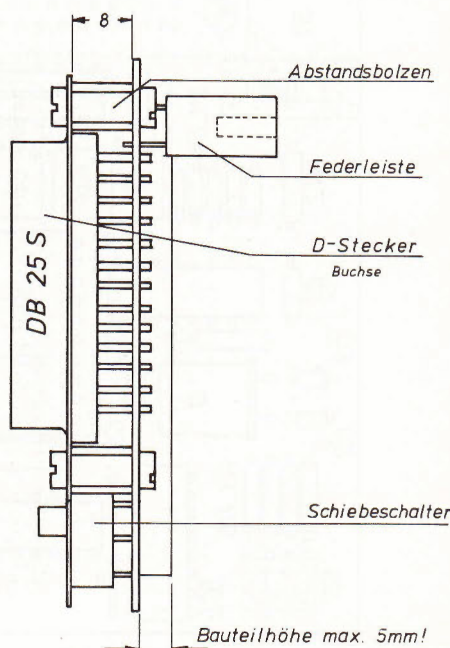
1. IC2 HEF 4060 (Taktgenerator).
An IC1 Pin 3, 4 muß eine Rechteckspannung mit der Frequenz 19.2 kHz (S1a) bzw. 38.4 kHz (S1b) anstehen.

2. IC3 74LS138 (Adressdecoder):
Pin 9/IC1 muß bei offenen Eingängen auf H-Pegel liegen. Legt man Pin 4, 5/IC3 auf GND, wechselt Pin 9/IC1 auf L-Pegel.

3. IC5 74LS85 (Adressdecoder), (4-Bit Komparator):
Pin 14/IC1 muß auf H-Pegel liegen, wenn die 4-Bit-Kombination an Pin 10, 12, 13,15/IC5 der 4-Bit-Kombination am Schalter S2 entspricht. Bei Ungleichheit wechselt Pin 14/IC1 auf L-Pegel.

4. IC4 MAX 232 oder ICL 232 (V-24-Treiber):
An Pin 2/IC4 muß eine Spannung von ca. +10 V und an Pin 6/IC4 eine Spannung von ca. -10 V meßbar sein.

5. IC1 MC 6850 (ACIA):
Die Überprüfung dieses ICs kann nur am Rechner erfolgen. Vorausgehen sollte jedoch eine sorgfältige optische Kontrolle der gesamten Platine auf



```
10 '********************
20 '*   RS-232-TEST   *
30 '********************
40 MODE 1:PAPER 0: PEN 1
50 PRINT "                RS-232            "
60 PRINT "______________________________________"
70 PRINT:PRINT:PRINT
80 PRINT"Baudrate:  300 Bd [ 600 Bd]  a)     S1a
90 PRINT"          1200 Bd [2400 Bd]  b)     S1b
100 PRINT:PRINT:INPUT"Welche Baudrate ";B$
110 IF b$="a" THEN BD=2 : GOTO 130
120 IF b$="b" THEN BD=1 ELSE GOTO  100
130 LOCATE 1,18:PRINT" TxD und RxD kurzschliessen
140 PRINT" RTS und CTS kurzschliessen
150 PRINT:PRINT
160 INPUT " <ENTER> ";E$
170 PAPER 5: PEN 3
180 OUT &F8DC,3
190 OUT &F8DC,16+BD
200 CLS
210 PRINT CHR$(24);"              RS-232 Test              ";CHR$(24)
220 IF INP(&F8DC) AND 8 THEN PRINT"CTS und RTS nicht kurzgeschlossen":END
230 FOR a=32 TO 255
240 IF INP(&F8DC) AND 2 THEN OUT &F8DD,a: GOTO 250 ELSE 240
250 IF INP(&F8DC) AND 1 THEN GOTO 260 ELSE 250
260 b=INP (&F8DD)
270 PRINT " "; CHR$(a);CHR$(24);CHR$(b);CHR$(24);" ";
280 IF a<>b THEN PRINT CHR$(7);"Fehler !";BIN$(INP(&F8DC),8):END
290 NEXT
300 LOCATE 1,1
310 END
```

Listing: ein kleines Testprogramm

Kurzschlüsse bzw. Unterbrechungen der Leiterbahnen und auf Lötschlüsse. Zusätzlich kann außerdem die Gesamtstromaufnahme gemessen werden, die ca. 100 - 120 mA betragen darf.

Funktionstest der Gesamtschaltung

Zunächst wird die Schnittstelle adressiert. Das kurze Testprogramm verwendet die Adresse F8DCh, d.h. S2a, b, c werden geschlossen, S2d bleibt geöffnet. Die Leitungen $\overline{\text{TxD}}$ und $\overline{\text{RxD}}$ sowie RTS und CTS werden an der D-Buchse kurzgeschlossen. Der gesamte Zeichensatz wird nun in einer Schleife seriell über $\overline{\text{TxD}}$ gesendet und über $\overline{\text{RxD}}$ wieder seriell in den Rechner eingelesen. Auf dem Bildschirm werden die empfangenen Zeichen invers dargestellt und mit den ausgesendeten verglichen. Ein Aufheben des Kurzschlusses zwischen CTS und RTS bewirkt eine Unterbrechung der Übertragung.

Bauteile-Liste

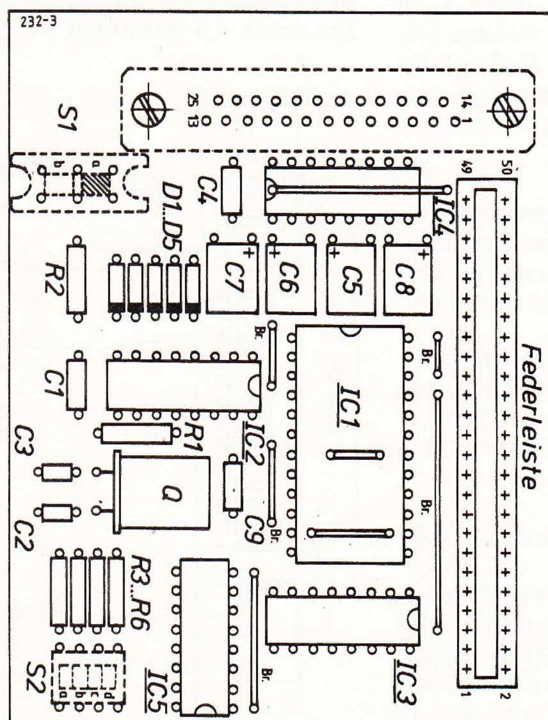
IC1	MC 6850
IC2	HEF 4060 B
IC3	74LS138
IC4	MAX 232
IC5	74LS85
C1,C4,C9	0,1 uF
C2	33pF
C3	100pF
C5-C8	22uF/16V (1uF..47uF möglich)
R1	1 M
R2	10k
R3-R6	4,7k
D1-D5	4148 o.ä.
Q	Quarz 4 MHz
S1	2xUM Schiebeschalter
S2	DIP-Schalter 4pol.
D-Stecker	DB-25-S (Buchse) 25 polig
Federleiste	(Slot) 2x25pol. RM 2,54

Bezugsquelle für MAX 232:
SE-Elektronik,
3062 Bückeburg
Postfach 13

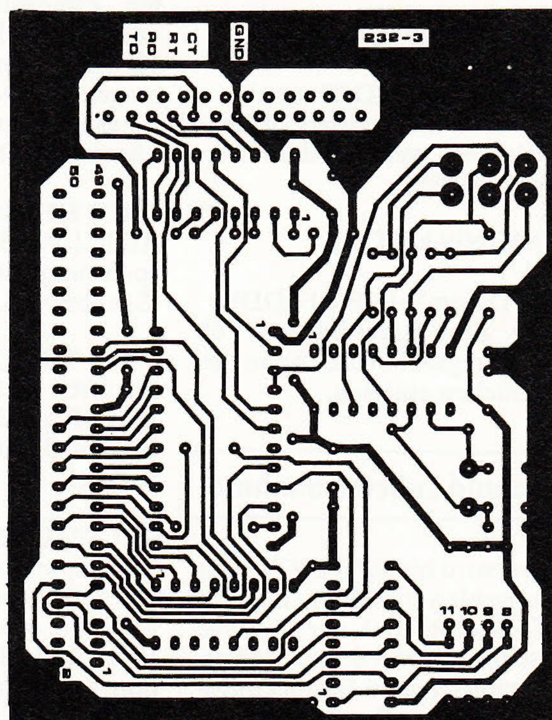
Literaturhinweise
Datenblatt MC 6850
Fa. MOTOROLA
Datenblatt MAX 232
Fa. SE-Elektronik
Datenbuch "LOCMOS-Reihe"
Fa. Valvo
TTL-Data-Book
Fa. Texas Instruments

(Joachim Schweda/ME)

Skizze: Bestückung der RS 232



Das Platinenlayout



CPC-Listingservice für 464-664-6128

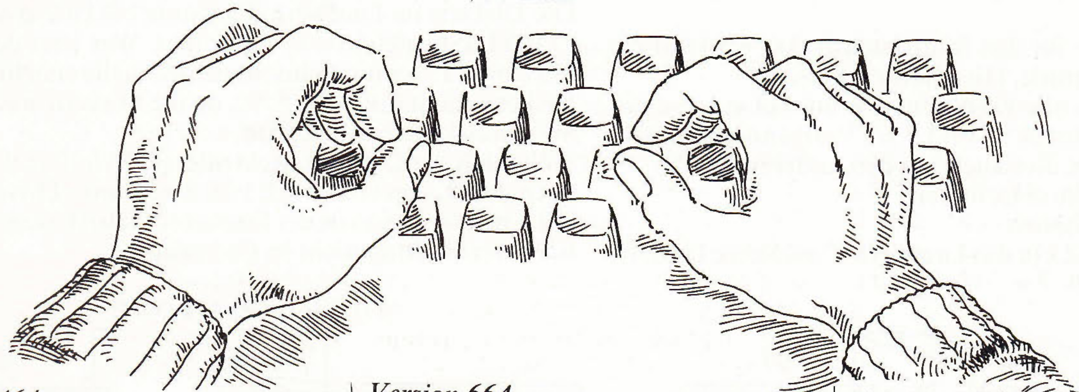


Wie in jedem Heft, finden Sie auch hier wieder den Listingservice, der Ihnen helfen soll, die in unserem Magazin abgedruckte "Paperware" in "Software" für Ihren Rechner zu verwandeln. Neben den Piktogrammen für Kassetten-/Diskettenbetrieb und der Angabe der Lauffähigkeit finden Sie die Super-Eintipphilfe "Checksummer". Um den Checksummer benutzen zu können, genügt es, das Programm einmal aufzurufen; es generiert selbständig den Maschinencode, aktiviert den Checksummer, gibt eine Bereitschaftsmeldung aus und löscht sich anschließend selbst wieder. Ab jetzt wird jede Eingabe, nachdem man ENTER gedrückt hat, mit der Ausgabe einer Quersumme quittiert. Beim Eintippen von Programmlistings kann man nun durch den Vergleich der beiden in eckigen Klammern stehenden Zahlen feststellen, ob man sich vertippt hat und ggf. den Fehler sofort berichtigen. Des weiteren hat man die Möglichkeit, mit dem RSX-Befehl CHECK,s,z ein Listing mit den dazugehörigen Checksummen zu erzeugen. Die Parameter s und z definieren den Ausgabekanal (s = 0 bis 9) und die Zeilennummer, ab der mit der Ausgabe begonnen werden soll; wird z nicht angegeben, so wird an dessen Stelle 1 angenommen.

Vor dem Druck sollte man mit WIDTH x die maximale Breite (x) einer Zeile festlegen. Mit Hilfe der RSX-Befehle ON und OFF kann der Cheksummer ein- bzw. ausgeschaltet werden.

Bei der Berechnung der Quersumme werden die Zeilennummer und Leerzeichen überlesen; zwischen Groß- und Kleinschreibung wird nur innerhalb zweier Anführungsstriche unterschieden, da ansonsten die Schreibweise von Befehlsworten und Variablennamen keinen Einfluß auf die Funktionstüchtigkeit eines Basicprogrammes hat.

(Thomas Fippl)



Version 464

```
100 REM *** CHECKSUM 464 ***
110 MEMORY &A4FF
120 FOR a%=&A500 TO &A607
130 READ byte$
140 POKE a%,VAL("&"+byte$)
150 NEXT
160 PRINT
170 PRINT"CHECKSUM v2"
180 CALL &A500:¦ON:NEW
190 :
200 DATA 21,09,a5,01,0d,a5,c3,d1
210 DATA bc,00,00,00,00,18,a5,c3
220 DATA 2a,a5,c3,2f,a5,c3,43,a5
230 DATA 4f,ce,4f,46,c6,43,48,45
240 DATA 43,cb,00,00,cf,98,aa,c3
250 DATA a8,a5,21,27,a5,18,03,21
260 DATA 24,a5,28,06,cd,00,b9,c3
270 DATA 06,dd,11,3a,bd,01,03,00
280 DATA ed,b0,c9,4f,cd,00,b9,0d
290 DATA 28,08,0d,20,ea,dd,7e,02
300 DATA 18,04,7b,11,01,00,cd,a2
310 DATA c1,cd,a3,e7,e5,4e,23,46
320 DATA 23,5e,23,56,e1,78,b1,c8
330 DATA cd,3c,c4,e5,09,e3,cd,63
340 DATA e1,21,a4,ac,cd,7a,a5,e1
350 DATA 18,e2,e5,cd,ba,a5,e3,cd
360 DATA 98,a5,cd,96,f2,e3,cd,f6
370 DATA a5,cd,4e,c3,e1,7e,a7,c8
380 DATA cd,98,a5,cd,4e,c3,18,f5
390 DATA 3a,24,ac,d6,08,47,7e,a7
400 DATA c8,cd,45,e1,23,10,f7,c9
410 DATA cd,24,a5,f5,c5,d5,e5,cd
420 DATA ba,a5,cd,f6,a5,e1,d1,c1
430 DATA f1,c9,eb,1b,af,47,67,6f
440 DATA 2f,32,23,a5,13,1a,d6,30
450 DATA 38,04,fe,0a,38,f6,1a,13
460 DATA a7,c8,4f,3a,23,a5,a1,fe
470 DATA 20,28,f3,79,fe,22,20,07
480 DATA 3a,23,a5,2f,32,23,a5,3a
490 DATA 23,a5,a7,79,c4,8a,ff,4f
500 DATA ad,07,6f,09,18,d8,3e,20
510 DATA cd,5c,c3,3e,5b,cd,5c,c3
520 DATA cd,79,ee,3e,5d,c3,5c,c3
```

Version 664

```
100 REM *** CHECKSUM 664 ***
110 MEMORY &A4FF
120 FOR a%=&A500 TO &A607
130 READ byte$
140 POKE a%,VAL("&"+byte$)
150 NEXT
160 PRINT
170 PRINT"CHECKSUM v2"
180 CALL &A500:¦ON:NEW
190 :
200 DATA 21,09,a5,01,0d,a5,c3,d1
210 DATA bc,00,00,00,00,18,a5,c3
220 DATA 2a,a5,c3,2f,a5,c3,43,a5
230 DATA 4f,ce,4f,46,c6,43,48,45
240 DATA 43,cb,00,00,cf,02,ac,c3
250 DATA a8,a5,21,27,a5,18,03,21
260 DATA 24,a5,28,06,cd,00,b9,c3
270 DATA 4c,cb,11,5b,bd,01,03,00
280 DATA ed,b0,c9,4f,cd,00,b9,0d
290 DATA 28,08,0d,20,ea,dd,7e,02
300 DATA 18,04,7b,11,01,00,cd,a9
310 DATA c1,cd,69,e8,e5,4e,23,46
320 DATA 23,5e,23,56,e1,78,b1,c8
330 DATA cd,75,c4,e5,09,e3,cd,59
340 DATA e2,21,8a,ac,cd,7a,a5,e1
350 DATA 18,e2,e5,cd,ba,a5,e3,cd
360 DATA 98,a5,cd,58,f3,e3,cd,f6
370 DATA a5,cd,9b,c3,e1,7e,a7,c8
380 DATA cd,98,a5,cd,9b,c3,18,f5
390 DATA 3a,09,ac,d6,08,47,7e,a7
400 DATA c8,cd,22,e2,23,10,f7,c9
410 DATA cd,24,a5,f5,c5,d5,e5,cd
420 DATA ba,a5,cd,f6,a5,e1,d1,c1
430 DATA f1,c9,eb,1b,af,47,67,6f
440 DATA 2f,32,23,a5,13,1a,d6,30
450 DATA 38,04,fe,0a,38,f6,1a,13
460 DATA a7,c8,4f,3a,23,a5,a1,fe
470 DATA 20,28,f3,79,fe,22,20,07
480 DATA 3a,23,a5,2f,32,23,a5,3a
490 DATA 23,a5,a7,79,c4,ab,ff,4f
500 DATA ad,07,6f,09,18,d8,3e,20
510 DATA cd,a3,c3,3e,5b,cd,a3,c3
520 DATA cd,49,ef,3e,5d,c3,a3,c3
```

Version 6128

```
100 REM *** CHECKSUM 6128 ***
110 MEMORY &A4FF
120 FOR a%=&A500 TO &A607
130 READ byte$
140 POKE a%,VAL("&"+byte$)
150 NEXT
160 PRINT
170 PRINT"CHECKSUM v2"
180 CALL &A500:¦ON:NEW
190 :
200 DATA 21,09,a5,01,0d,a5,c3,d1
210 DATA bc,00,00,00,00,18,a5,c3
220 DATA 2a,a5,c3,2f,a5,c3,43,a5
230 DATA 4f,ce,4f,46,c6,43,48,45
240 DATA 43,cb,00,00,cf,02,ac,c3
250 DATA a8,a5,21,27,a5,18,03,21
260 DATA 24,a5,28,06,cd,00,b9,c3
270 DATA 49,cb,11,5e,bd,01,03,00
280 DATA ed,b0,c9,4f,cd,00,b9,0d
290 DATA 28,08,0d,20,ea,dd,7e,02
300 DATA 18,04,7b,11,01,00,cd,a6
310 DATA c1,cd,64,e8,e5,4e,23,46
320 DATA 23,5e,23,56,e1,78,b1,c8
330 DATA cd,72,c4,e5,09,e3,cd,54
340 DATA e2,21,8a,ac,cd,7a,a5,e1
350 DATA 18,e2,e5,cd,ba,a5,e3,cd
360 DATA 98,a5,cd,53,f3,e3,cd,f6
370 DATA a5,cd,98,c3,e1,7e,a7,c8
380 DATA cd,98,a5,cd,98,c3,18,f5
390 DATA 3a,09,ac,d6,08,47,7e,a7
400 DATA c8,cd,1d,e2,23,10,f7,c9
410 DATA cd,24,a5,f5,c5,d5,e5,cd
420 DATA ba,a5,cd,f6,a5,e1,d1,c1
430 DATA f1,c9,eb,1b,af,47,67,6f
440 DATA 2f,32,23,a5,13,1a,d6,30
450 DATA 38,04,fe,0a,38,f6,1a,13
460 DATA a7,c8,4f,3a,23,a5,a1,fe
470 DATA 20,28,f3,79,fe,22,20,07
480 DATA 3a,23,a5,2f,32,23,a5,3a
490 DATA 23,a5,a7,79,c4,ab,ff,4f
500 DATA ad,07,6f,09,18,d8,3e,20
510 DATA cd,a0,c3,3e,5b,cd,a0,c3
520 DATA cd,44,ef,3e,5d,c3,a0,c3
```

5 ¼"-Laufwerk am CPC 464 ohne Kompromisse

Im Sonderheft 2/86 CP/M - Floppys - Hardware wurde der Anschluß von 5 1/4"-Laufwerke insbesonders am CPC 664 eingehend behandelt. Da nicht jeder Schaltunterlagen des CPC 464 besitzt, wird nachfolgend der Umbau genauer beschrieben. Zum Umbau wird allerdings das Sonderheft benötigt, da auf die dort veröffentlichten Änderungen an den Laufwerken bzw. den Anschlußbelegungen hier nicht eingegangen wird.
Wer keine Änderung an der Controllerplatine vornehmen will, benötigt lediglich ein Kabel nach Skizze A).
Ich mache ausdrücklich darauf aufmerksam, daß nach Beendigung der Arbeit die Verbindungen auf Durchgang zu prüfen sind. Wird das Kabel wie auf der Skizze angegeben angefertigt, bringt man die Kabel farblich auf Deckung, da der selbstgebaute Stiftleistenstecker leider verwechselbar ist.
(Als Bezugsquelle für den Stiftleistenstecker wird auf die Fa. Conrad Electronic, Hirschau verwiesen.)
Prinzipiell dürfen alle Verbindungen nur bei ausgeschaltetem Gerät, Laufwerk 3" und 5 1/4" vorgenommen werden. Natürlich gilt dies auch bei den anderen CPC's.
a) Floppylaufwerke einschalten
b) CPC 464 einschalten
c) Diskette CP/M 2.2 in das Laufwerk 3" und leere Diskette 5 1/4" in das 5 1/4"-Laufwerk schieben und nachfolgende Eingaben vornehmen:

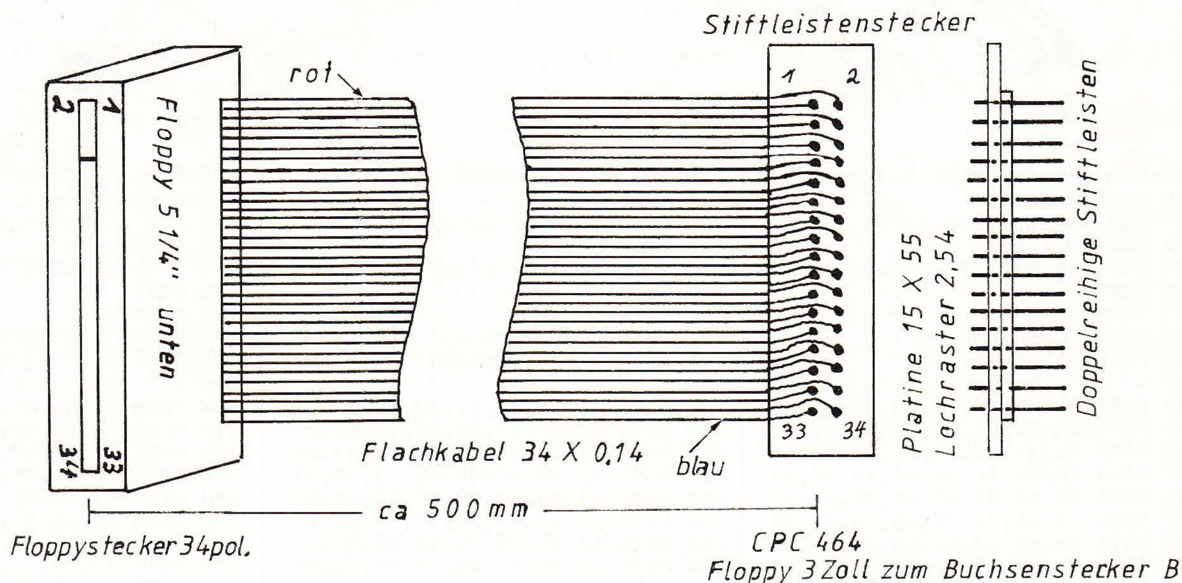
I CPM
CP/M 2.2 - Amstrad Consumer Electronics plc

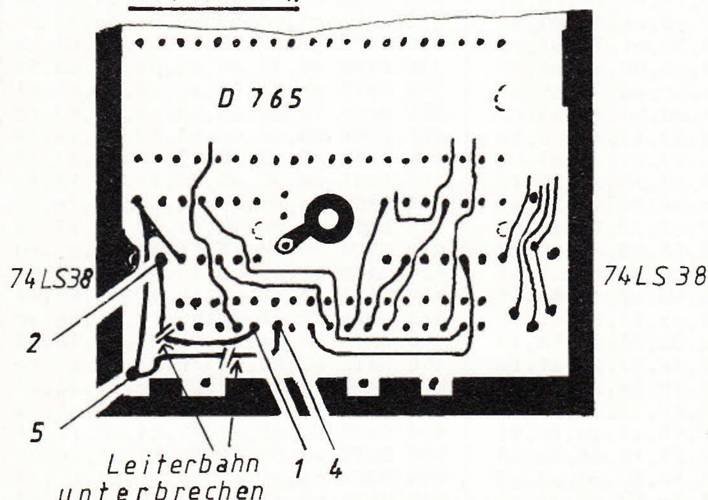
A>copydisc

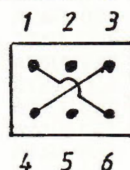
COPYDISC 2.0
Please insert source disc into drive A and destination disc into drive B then press any key:_
copying started
Reading track 1 ((...8) usw. (bis 40))
Writing track 1 ((...8) usw. (bis 40))
Copying complete
Do you want to copy another disc(Y/N):_n

A>b:
B>era *.*

Die Diskette im Laufwerk B> wurde bei Übertragung mit COPYDISC gleichzeitig formatiert. Wer jetzt den weiteren Umbau vornimmt, löscht natürlich die erstellte CP/M-Diskette nicht mit >era *.*<, da die Diskette nachher als Systemdiskette benutzt wird.
Sollte sich der CPC gar nicht oder fehlerhaft melden bzw. nach dem Lesen von Track 1 bis 8 sich mit - Drive B: write fail ... melden, dann ist der Stecker verkehrt herum bzw. die Kabelverbindung nicht in Ordnung.





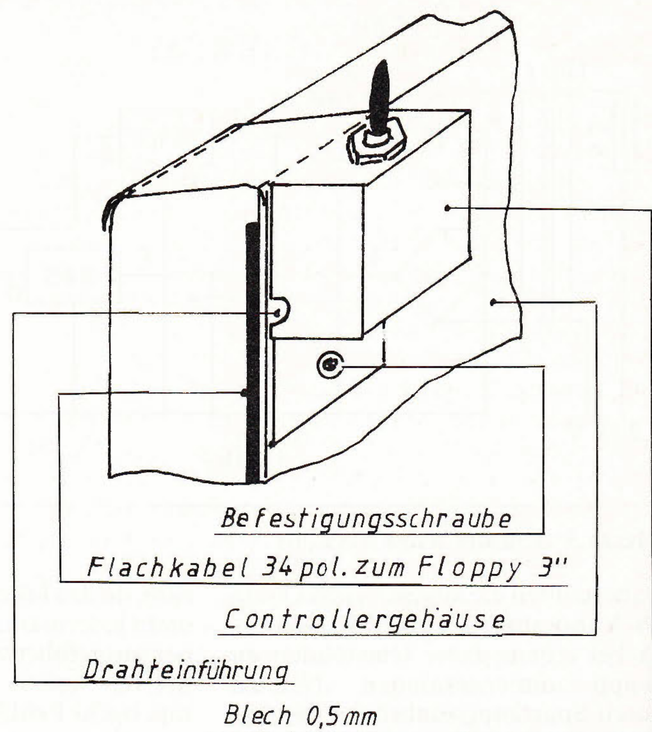


Zum weiteren Umbau werden nun die beiden Schrauben auf der Rückseite des Controllergehäuses gelöst. Danach werden die beiden Leiterbahnen laut Skizze B). unterbrochen. An die Punkte 2, 5, 1 und 4 werden ca. 60 mm lange isolierte Drähte angelötet, die danach entsprechend mit den jeweiligen Punkten am Miniaturschalter verbunden werden. Am Schalter werden ebenfalls über Kreuz zwei Stück isolierte Drähte angelötet.

Auf Skizze C) ist ein Anbauvorschlag zu sehen, wobei keine Änderung am Kunststoffgehäuse vorgenommen wird. Der Berührungsschutz am Schalter wird aus Weißblech (Dosenblech) 0.5 mm angefertigt. Die Größe richtet sich je nach Miniaturschalter. Oben läßt man das Blech ca. 30 mm länger. Das Blech läßt sich sehr gut zusammenlöten. Danach bohrt man ein 3 mm Loch für die Befestigung, und je nach Schalter (in der Regel 6 mm) ein Loch im oberen Teil. Nun wird das Blech sauber über das Kunststoffgehäuse gebogen. Mit einer Flachzange etwas nachbiegen, Schalter einbauen und zusammen mit dem Kunststoffdeckel des Controllergehäuses verschrauben.

Mit diesem Umbau ist man in der Lage, je nach >Schalterstellung< das Floppy 5 1/4" als Laufwerk A und das 3" als Laufwerk B zu definieren.
Wählt man das Floppy 5 1/4" als A, wird nun wie beim 3"-Floppy die vorher erstellte Systemdiskette 5 1/4" eingeschoben und mit **I** cpm gestartet. Anschließend A format eingeben und die übrigen 5 1/4"-Disketten formatieren.
Wer letztgenannten Umbau scheut und sich lediglich das Kabel anfertigt, kann sich seine Disketten nur mit >copydisk vom 3"-Laufwerk formatieren.

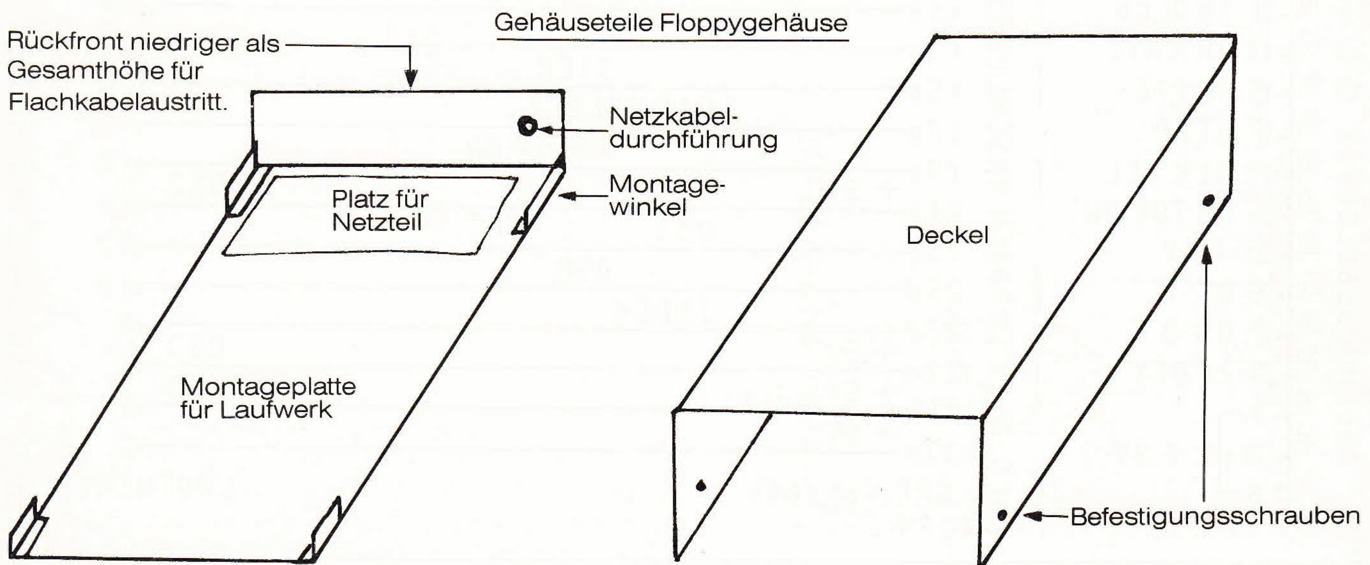
(Hans Ziegler)



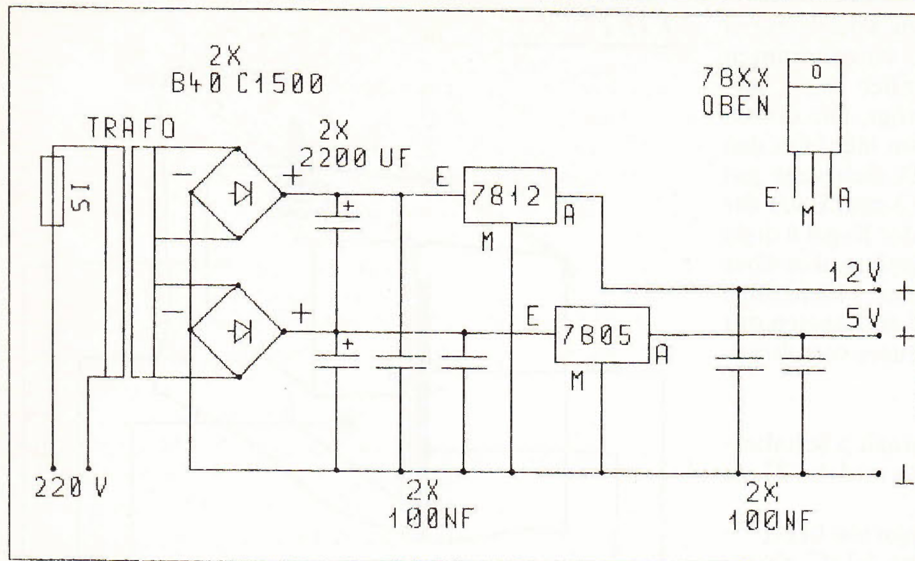
Zweitlaufwerk im Selbstbau - 3"-Floppy für ca. 200,-DM

„Das darf doch nicht wahr sein!" - das war meine Reaktion auf eine Annonce in einer Computerzeitschrift, in der von einem bekannten Elektronikversandhaus ein 3"-Diskettenlaufwerk für unter 100,- DM angeboten wurde. Ein solches fehlte mir gerade noch zur Komplettierung des Computerarbeitsplatzes.
Es handelte sich um den Restposten eines Hitachi 'HFD 305 S'-3"-single-sided-Laufwerks. Da das Laufwerk mit dem sogenannten "Shugart-Bus" ausgestattet ist, ein READY-Signal zur Verfügung steht und die Kopftransportzeit von Spur zu Spur mit 3 mS angegeben wird, sollte der Zusammenarbeit mit den CPC's nichts mehr im Weg stehen.
Der Einbau des Laufwerks jedoch gestaltet sich recht schwierig, da auf dem Markt kaum fertige Gehäuse für 3"-Laufwerke erhältlich sind. Man kann sich behelfen, indem man Gehäuse von 3,5"-Floppys verwendet und sie an der Frontseite mit einer Blende versieht, die die Aussparungen der 3"-Laufwerke aufweist. Eine andere Lösung ist der Selbstbau eines Gehäuses aus zwei U-förmig gebogenen Gehäusehälften, die mittels Winkelschienen miteinander verbunden werden.

Als Netzteil kann ein handelsübliches dienen, welches mindestens 12V/1A und 5V/0,8A liefern können muß, oder man baut sich nach Skizze 3 ein eigenes Netzteil auf.



Maße sind je nach Netzteil zu ermitteln vom "Maß-Schneider"



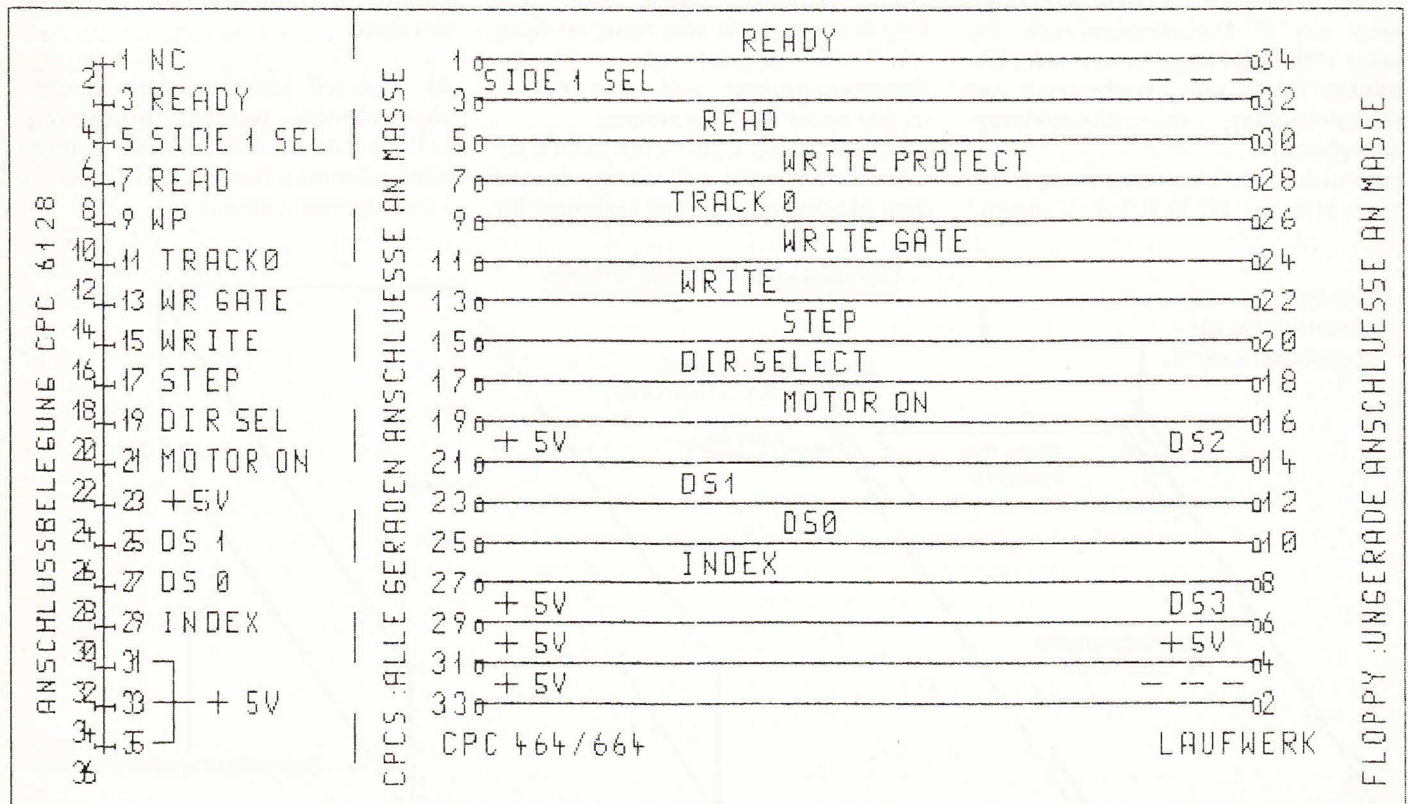
Skizze 3: Schaltbild des Netzteils

Dabei sollten die angegebenen Daten als Mindestwerte aufgefaßt werden, da bei einem unter Umständen zu knapp dimensionierten Netzteil durch Spannungseinbrüche die Anzahl fehlerhafter Diskettenoperationen deutlich ansteigen wird.

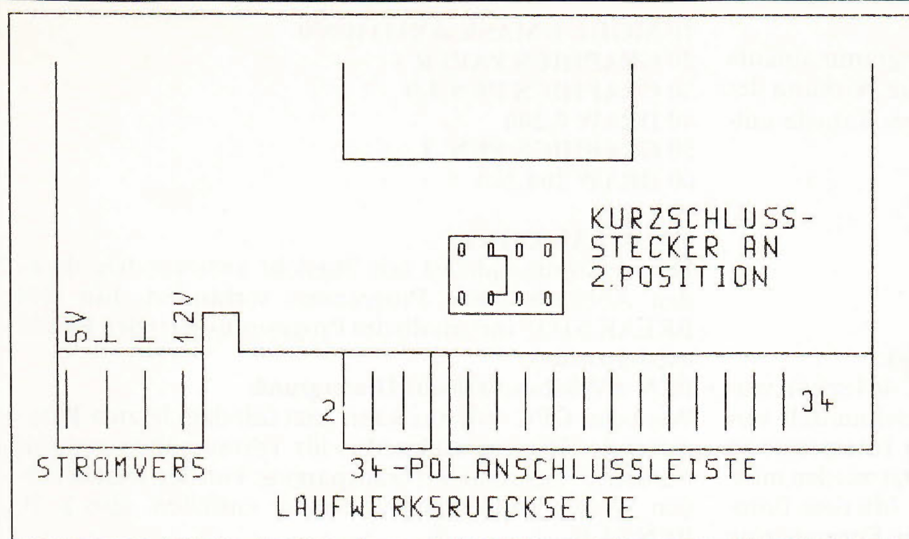
Für die Herstellung des Anschlußkabels zum Computer wird floppyseitig ein 34-poliger 'card-edge'-Stecker benötigt. Den gleichen Stecker braucht man zum Anschluß an CPC 464/664; für den 6128 wird ein 36-poliger 'Amphenol'-Stecker benötigt. Diese Stecker besorgt man am besten in der sogenannten "Schneidklemm"-Version, da das Löten auf engstem Raum nicht jedermanns Sache ist, aber sauber ausgeführt werden muß, da sich bei der hohen Datenübertragungsrate leicht Fehlfunktionen einschleichen können. Bei der Schneidklemm-Ausführung ist die Montage des Flachbandkabels recht einfach: Man legt das Kabel in die Vertiefungen der einen Steckerhälfte und drückt den Stecker zunächst mit der Hand zusammen, bis die Messer der anderen Steckerhälfte sich in das Kabel eingraben. Anschließend wird der Stecker in einem Schraubstock vollständig zusammengedrückt. Dabei ist darauf zu achten, daß der Anschluß für den Shugart-Bus am CPC spiegelbildlich zu dem des Laufwerks ist (siehe Skizze 1 und Computerhandbuch). Nachdem das Flachbandkabel montiert ist, muß jetzt das Laufwerk noch mit Energie versorgt werden. Die Anschlußpunkte für die Spannungsversorgung sind auf der Platine des Laufwerkes vor der Buchse für die Stromversorgung aufgedruckt.

Nachdem nun das Flachkabel angeschlossen wurde (auf seitenrichtigen Anschluß achten!), steht einem ersten Test des Laufwerks nichts mehr im Wege. Das Laufwerk muß sich jedoch frei bewegen können (Unterseite beachten!). Wenn jetzt das Laufwerk angesprochen wird, sollte sich eine Reaktion zeigen. Erscheint aber eine Fehlermeldung (Drive B: Disc missing), so ist wahrscheinlich der Kurzschlußstecker, mit dem auf der Platine die Laufwerksadresse eingestellt wird, nicht richtig gesetzt. Dieser Stecker sollte im Anschlußfeld rechts oberhalb des Busanschlusses an der zweiten Stelle von links sitzen (siehe Skizze 2).

Wenn die ersten Tests positiv verlaufen sind, und das Laufwerk B in allen Funktionen korrekt arbeitet, kann der Einbau in ein passendes Gehäuse erfolgen. Hierbei muß wiederum auf eine freie Beweglichkeit aller Laufwerksteile geachtet werden.



Skizze 1: Anschlußschema des Kabels



Skizze 2: Anschlußseite der Platine

Nachdem die Stromversorgung und das Flachbandkabel angeschlossen wurden, steht einem Betrieb als Zweitlaufwerk nichts mehr im Weg. Wenn ein Metallgehäuse zum Einbau des Laufwerks verwendet wird, muß dieses mit dem Schutzleiter (grün/gelb) des Netzkabels verbunden werden.
Das genannte Laufwerk ist u.a. auch zu beziehen von der Firma Völkner elektronic 3300 Braunschweig

Bauteileliste Laufwerk:

Laufwerk Hitachi HFD 305 S
40 cm Flachbandkabel 34-polig
Card-edge-Stecker 34-polig für Laufwerk
Card-edge-Stecker 34-polig oder Centronics-Stecker 36-polig für Computer
Stecker 4-pol.für Spannungsversorgung
Gehäuse für Laufwerk
Netzteil 12V/1A und 5V/0,8A

Bauteileliste Netzteil:

Transformator 2 x 9V/1A
2 Gleichrichter B40 C1500
2 Elkos 2200uF/25V
4 Kondensatoren 100 nF
Spannungsregler 78S05
Spannungsregler 78S12
Sicherungshalter
Feinsicherung 0,125 A

Hinweise zum Aufbau des Netzgerätes:
Die verwendeten Spannungsregler sollten zur Wärmeableitung isoliert auf einer größeren Metallfläche montiert werden· die thermische Verbindung sollte mit Wärmeleitpaste unterstützt werden.

Die Ausgangsspannung des Trafos sollte 2 x 10 V nicht überschreiten, da der 12V-Regler nur eine Eingangsspannung von max. 30V verträgt, die bei mehr als 2 x 10V im Leerlauf schnell überschritten werden kann.

Ein letzter Hinweis:
Die Anschlußbelegung der Zweitlaufwerksbuchse am 6128 ist im Handbuch falsch beschriftet; in Skizze 1 ist die korrekte Belegung angegeben.

(W. Wantia/ME)

HARDWARE

664/6128-Emulator für den CPC 464

Nicht nur für CPC 464-Besitzer ist das eine frohe Botschaft - auch 664/6128-Programmierer können jetzt hemmungslos ihr erweitertes Basic nutzen, ohne die Inkompatibilität zum "kleinen Bruder" fürchten zu müssen.

Das folgende Programm installiert auf dem CPC 464 weitgehend die zusätzlichen Fähigkeiten des Basic 1.1. Schreiben Sie zunächst ein kleines Ladeprogramm,

```
10 SYMBOL AFTER 256
20 MEMORY &A1FF
30 SYMBOL AFTER 240
40 LOAD "EMU.BIN"
50 CALL &A200
60 PRINT"BASIC 1.1 installiert"
```

und speichern Sie es auf Diskette/Kassette ab. Danach wird der Basiclader abgetippt und insbesondere bei Benutzung des Checksummers **vor dem Start** zur Sicherheit abgespeichert! Er erzeugt das Maschinenprogramm "EMU.BIN", das bei Kassettenbetrieb natürlich direkt hinter das Ladeprogramm geschrieben werden muß. Wie Sie sehen werden, ist das Programm erstaunlich kurz - es beansprucht noch nicht einmal einen Block auf Kassette und ist deshalb bei Bedarf sehr schnell geladen.
Die neuen Befehle wurden nicht per RSX, sondern als "echte" Kommandos und Funktionen eingebunden, wobei bis auf eine unvermeidliche Ausnahme auch die gleiche interne Codierung wie beim CPC 664/6128 verwendet wurde. Deshalb können auf diesen Rechnern entwickelte Programme ohne Probleme von Kassette/Diskette auf den CPC 464 übernommen werden - bei Programmen im ASCII-Format funktioniert das auf jeden Fall; ansonsten kann in wenigen Fällen eine kleine Anpassung notwendig werden, die gleich noch zur Sprache kommt.
Hier nun die neuen bzw. veränderten Befehle mit ihrer Syntax:

CLEAR INPUT:
Dieses Kommando löscht den Tastaturpuffer und verhindert, daß während eines Programmablaufs eingegebene Zeichen ungewollt durch INPUT oder INKEY$ übernommen werden.
Dieser Befehl löst auf dem CPC 464 einen Syntax-Error aus, wenn er in einem 664/6128-Programm vorkommt, welches nicht im ASCII-Format (SAVE "Name",A) aufgezeichnet wurde. Wie gewohnt wird dabei nach der Fehlermeldung die entsprechende Zeile auf dem Bildschirm editiert. Drücken Sie in diesem Fall einfach ENTER, um die Zeile **unverändert** wieder einzugeben, und starten Sie das Programm neu - der Emulator sorgt automatisch für eine Korrektur. Soll ein mit dem CPC 464 plus Emulator geschriebenes Programm, das diesen Befehl enthält, auch auf einem CPC 664/6128 laufen, so muß es auf jeden Fall im ASCII-Format aufgezeichnet werden.
COPYCHR$(#Window):
Diese Funktion liest das Zeichen an der aktuellen Cursorposition im angegebenen Textfenster:

```
10 LOCATE 10,10:PRINT "X"
20 LOCATE 10,10:a$=COPYCHR$(#0)
30 PRINT a$
```

CURSOR Schalter1,Schalter2:
Damit kann der Cursor während eines Programmablaufs ein- oder ausgeschaltet werden. Die genaue Wirkung der beiden Schalter können Sie der folgenden Tabelle entnehmen:

CURSOR 0,0 aus/aus bei INPUT
CURSOR 0,1 aus/ein bei INPUT
CURSOR 1,0 aus/aus bei INPUT
CURSOR 1,1 ein/ein bei INPUT

DEC$ (numerischer Ausdruck,Formatstring):
Diese Funktion ist eigentlich auf dem CPC 464 schon vorhanden, wurde aber im Anleitungsbuch schamhaft verschwiegen, da aufgrund eines Fehlers im Interpreter zu Beginn immer zwei offene Klammern gesetzt werden müssen, um einen Syntax-Error zu vermeiden. Mit dem Emulator funktioniert es aber nun richtig: Der Formatstring wird wie bei PRINT USING angegeben, und der numerische Ausdruck damit in eine entsprechend formatierte Zeichenkette umgewandelt.

```
a$=DEC$(PI,"#.###"):PRINT a$
```

FILL Farbstift:
Ein schnelles Fill für den CPC 464 wurde bereits in CPC International vorgestellt, doch die 664/6128-Version hat ihre besonderen Tücken: Als Umrandung oder Begrenzung (Sperrfarbe) gelten nur Linien, die mit dem aktuellen GRAPHICS PEN oder der Füllfarbe gezeichnet wurden. Auf diese Weise können sogar mehrfarbige Flächen ausgemalt werden, aber man muß schon sehr aufpassen, ob die Begrenzung wirklich in der gewünschten Weise als Sperrfarbe akzeptiert wird. Der Füllvorgang beginnt immer an der aktuellen Position des Grafikcursors, er muß also vorher mit MOVE x,y in die Fläche gesetzt werden.

FRAME:
Synchronisiert die Bildschirmausgabe mit dem Strahlrücklauf des Monitors und unterstützt die wirklichkeitsgetreue Bewegung von Zeichen.

GRAPHICS PAPER Farbstift:
Bestimmt die Farbe des Grafik-Hintergrundes bei Textausgaben auf Grafikcursorposition (nach TAG) und bei Anwendung des MASK-Befehls.

GRAPHICS PEN Farbstift,Hintergrund:
Wählt die Schreibfarbe und den Hintergrundmodus für Grafik. Der zweite Parameter hat folgende Bedeutung:
0 = normal überschreiben
1 = transparent
Jeweils einer der beiden Parameter kann entfallen, wenn der Wert unverändert bleiben soll. Die Wirkung des bisher auf dem CPC 464 nicht vorhandenen Grafik-Transparentmodus demonstriert das folgende Beispiel:

```
10 MODE 1:TAG:GRAPHICS PAPER 2
20 GRAPHICS PEN 1,0
30 MOVE 100,100:PRINT "A";
40 GRAPHICS PEN 3,1
50 MOVE 108,108:PRINT "Z";
```

MASK Bitmaske:
Eine Bitmaske ist ein Wert von 0 - 255, der den Status von jeweils acht benachbarten Bildpunkten beim Zeichnen einer Linie bestimmt. Damit können beliebig unterbrochene bzw. gestrichelte Linien erzeugt werden. Die Zwischenräume werden mit der GRAPHICS PAPER-Farbe gefüllt oder bleiben unverändert, falls der Transparentmodus für Grafik gewählt wurde.

```
10 MODE 1:MASK &X11110000
20 GRAPHICS PAPER 3
30 GRAPHICS PEN 1,0
40 DRAW 0,200
50 GRAPHICS PEN ,1
60 DRAW 200,200
```

ON BREAK CONT:
Dieses Kommando ist mit Vorsicht anzuwenden, da es den Abbruch eines Programms verhindert. Ein ON BREAK STOP innerhalb des Programms setzt den Befehl wieder zurück.

PEN #Window,Farbstift,Hintergrund:
Wie beim CPC 464, nur kann jetzt mit dem letzten Parameter der Hintergrundmodus für Textausgaben gewählt werden (0 = normal, 1 = transparent). Von den letzten beiden Werten kann wahlweise einer entfallen, also z. B. PEN#1,,0

MERGE und CHAIN MERGE "Dateiname":
Diese Kommandos funktionieren jetzt auch bei Diskettenbetrieb einwandfrei. Die unerwünschte Fehlermeldung "EOF met" wird vom Emulator abgefangen.

Parameterübergabe bei CALL und RSX:
Bei CALL und RSX-Kommandos können Strings direkt ohne Umwege übergeben werden, also z. B. so:

≤ERA,"TESTPROG.BAK"

Zusätzliche Grafik-Parameter:
Alle Grafik-Befehle werden auf das folgende Format erweitert:

KOMMANDO x,y,Farbstift,Schreibmodus

Das gilt für die Befehle MOVE, MOVER, PLOT, PLOTR, DRAW, DRAWR. Der letzte Parameter gibt an, wie die Bildpunkte mit dem Hintergrund verknüpft werden sollen:
0 - normal überschreiben
1 - XOR-Modus
2 - AND-Modus
3 - OR-Modus
Die Verknüpfungsvorschrift wird auf die Farbstiftnummer der Schreibfarbe und des Hintergrundes angewandt.

Kleine Abstriche:
Wie bereits beim CLEAR INPUT-Kommando angedeutet, kann eine vollständige Kompatibilität zwischen den verschiedenen CPC-Rechnern nicht erreicht werden, da einige der Unterschiede unveränderlich in den ROM's verankert sind.

Für eine Realisierung der DERR-Funktion wurde bisher keine Möglichkeit gefunden. Sie wird zwar vom Emulator akzeptiert, gibt aber nur den Wert 0 zurück. Wenn ein Leser Rat weiß, soll er sich melden: Gebraucht wird eine Routine, die die Fehlernummer im Akku übergibt.
Der CPC 664/6128 erlaubt bei dem MASK-Befehl noch einen zweiten Parameter, der bestimmt, ob beim Zeichnen von Linien der erste Punkt gesetzt werden soll oder nicht, was im XOR-Modus mitunter nützlich sein kann. Der Emulator akzeptiert zwar diesen zusätzlichen Wert, führt das Kommando aber nicht aus, da der programmtechnische Aufwand für eine Realisierung in keinem Verhältnis zur Bedeutung des Kommandos steht.
Die zusätzlichen Betriebssystem-Vektoren (z.B. GRA FILL) des 664/6128 konnten leider nicht eingebaut werden, da in diesem Bereich die Arithmetik-Vektoren des CPC 464 liegen. Eine Kompatibilität auf Maschinenebene ist hier also nicht gewährleistet. 664/6128- Programmierer, die auf die Hilfe des Emulators bei Kompatibilitätsproble-

Software von Lesern – für Leser

DIE GOLDENE 7

NEU JETZT NR. 2

Schatz
Als Hubschrauberpilot für besonders brenzlige Einsätze bekommen Sie den Auftrag, durch ein verzwicktes Höhlensystem bis zu einem legendären Schatz vorzudringen und mit diesem wieder heil zurückzukehren.
Aber Vorsicht ist geboten, der Schatz ist schwer und die Höhlen verwinkelt und eng.

Garten Manager
Als Gärtner hat man es nicht leicht. Versorgen Sie das Grünzeug in Ihrem Garten richtig, dann kann gar nichts passieren. Abgesehen davon, daß Sie überall das Unkraut jäten und die Blumen gießen müssen, geht fast alles wie von selbst.
Doch das Unkraut wuchert wie wild und die Blumen brauchen immer mehr Wasser. Haben Sie einen grünen Daumen?

7 Programme für CPC 464/664/6128

Berg der Monster
Grauenerregende Ungeheuer terrorisieren einen abgelegenen Landstrich in den Bergen. Als Lehensherr dieser Provinz bekommen Sie vom König den Auftrag, wieder Ruhe und Ordnung zu schaffen. Als Sie sich zu Ihrer Mission aufmachen, werden Sie von einem Schneesturm überrascht. Nach dem Unwetter stehen Sie ohne Ausrüstung und Hilfe alleine inmitten der Berge.
Gelingt es Ihnen trotzdem, Ihren Auftrag zu erfüllen?

Poker
Kartenspieler hergehört: wollen Sie ganze Nächte durchzocken, ohne Muttis Haushaltsgeld zu verspielen? Bitte sehr, unser Computerpoker macht's möglich. Gewinnen können Sie zwar nichts, aber trainieren läßt sich mit diesem Spiel umso besser.

Super Chance
Verwandeln Sie Ihren CPC in einen einarmigen Banditen. Mit dieser Spielhallensimulation holen Sie sich ein Stück Las Vegas in Ihr Wohnzimmer. Mit Jackpot und Superchance.

Captain Starships Test
Begleiten Sie Captain Starlight auf seiner Weltraumexpedition. Doch zuerst müssen Sie beweisen, daß Sie die notwendigen Qualitäten mitbringen. Ein kniffeliger Intelligenz- und Reaktionstest steht Ihnen bevor. Sind Sie in der Lage zu bestehen?

Oil Willi
Sie sind der nette Heizölhändler, der alle Bewohner seines Stadtviertels mit Heizöl versorgt. Aber die meisten Bestellungen sind sehr eilig. Und wenn Sie nicht schnell genug liefern, so bestellen sich die Leute ihr Öl eben anderswo. Da kann man schon ganz schön ins Schwitzen geraten.

Die Bezeichnung »Goldene Sieben« steht künftig für gute Software, die von den Lesern unserer Zeitschrift erstellt worden ist, und nicht als Listing im Heft abgedruckt werden kann, da hierfür nur ein begrenzter Raum vorhanden ist, und Ihre "Schneider CPC International" kein reines Listingblättchen sein soll!
Jeweils sieben tolle Programme sind auf der Kassette bzw. Diskette für wenig Geld enthalten.

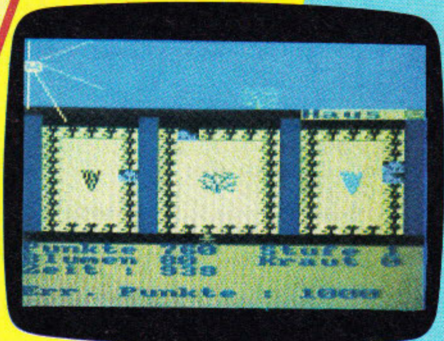
Preise:

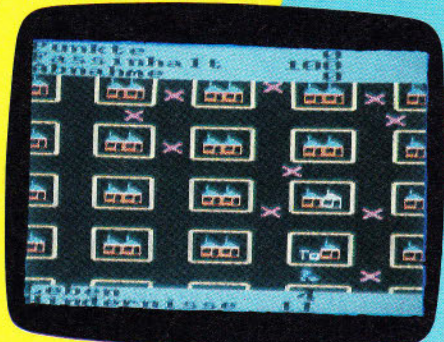
Kassettenversion	20,– DM
3"-Diskettenversion	30,– DM

zuzüglich 3,– DM Porto/Verpackung (Im Ausland 5,– DM)
Bei Nachnahme kommt noch die die Nachnahmegebühr hinzu.
Ihre Bestellung richten Sie bitte an:

DMV Verlag
– Leserservice –
Postfach 250
3440 Eschwege

Bitte Bestellkarte benutzen!







7 Programme für CPC 464/664/6128

Auch weiterhin ist die "Goldene 7" Nr. 1 mit den Programmen Secret of Wizard, Turlen, Zodora, 3-D Labyrinth, Die alte Burg, Space-Race und Galaxis erhältlich.
Bitte benutzen Sie die Bestellkarte!

men hoffen und Assemblerroutinen in ihr Programm einbinden wollen, sollten noch die folgenden Tatsachen berücksichtigen:

- Der Emulator belegt den Speicherbereich ab &A200
- Folgende Indirections wurden über neue Routinen geleitet:
  GRA LINE &BDE2
  SCR WRITE &BDE8
- Die Vektoren TXT SET PEN (&BB90), GRA SET PEN (&BBDE), GRA MOVE ABSOLUTE (&BBC0) und GRA MOVE RELATIVE (&BBC3) wurden verbogen, um die neue Parameterübergabe zu realisieren. Sie können trotzdem in gewohnter Weise benutzt werden, solange das obere ROM beim Aufruf ausgeschaltet ist.

Ebenfalls noch nicht geglückt ist die softwaremäßige Simulation der zusätzlichen 64 KByte des CPC 6128. Wir haben dieses Problem jedoch unserer Abteilung "Zeichen und Wunder" übergeben, so daß vielleicht in den nächsten Jahrzehnten mit einer Lösung zu rechnen ist...

(M. Uphoff)

```
10 '*********************************          [1460]
20 '* 664/6128 -Emulator fuer CPC 464 *        [1839]
30 '*********************************          [1460]
40 '******* (C) 1986  M. Uphoff *******        [1189]
50 '*********************************          [1460]

60 '                                          [117]
70 adr=&A1FF:MEMORY adr                       [1232]
80 FOR bl=1 TO 9                              [675]
90 s=0:READ sum                               [660]
100 FOR i=1 TO 128                            [554]
110 READ byte$:IF byte$="EOF" THEN 150        [1777]
120 v=VAL("&"+byte$):s=s+v                    [2224]
130 adr=adr+1:POKE adr,v                      [1023]
140 NEXT i                                    [375]
150 IF s<>sum THEN PRINT"Datafehler in Blo    [3802]
ck";bl:END
160 PRINT"Block";bl;"OK"                      [2167]
170 NEXT bl                                   [440]
180 SAVE"emu.bin",b,&A200,&439                [1427]
190 END                                       [110]

200 '                                         [117]
210 '*********************** Block 1          [1441]
220 DATA 14235                                [366]
230 DATA 3E,C3,32,16,AC,21,83,A2              [1312]
240 DATA 22,17,AC,32,19,AC,21,8E              [1277]
250 DATA A2,22,1A,AC,32,07,AC,21              [1313]
260 DATA 9A,A2,22,08,AC,32,04,AC              [1745]
270 DATA 21,B4,A4,22,05,AC,32,0D              [1184]
280 DATA AC,21,2A,A3,22,0E,AC,32              [1627]
290 DATA 01,AC,21,AB,A2,22,02,AC              [1019]
300 DATA 32,90,BB,21,D8,A2,22,91              [1575]
310 DATA BB,32,DE,BB,21,11,A3,22              [1810]
320 DATA DF,BB,32,C0,BB,21,F5,A2              [1232]
330 DATA 22,C1,BB,32,C3,BB,21,FA              [1543]
340 DATA A2,22,C4,BB,21,53,A3,22              [1537]
350 DATA E3,BD,21,62,A3,22,E9,BD              [2104]
360 DATA 3A,80,BC,FE,C3,C8,32,36              [1770]
370 DATA A6,2A,81,BC,22,37,A6,3E              [944]
380 DATA C3,32,80,BC,21,B2,A2,22              [1562]

390 '*********************** Block 2          [1448]
400 DATA 19616                                [211]
410 DATA 81,BC,C9,11,E1,A5,CD,27              [1497]
420 DATA E3,D0,F1,C3,61,DF,21,E1              [1285]
430 DATA A5,CD,13,E3,D0,7E,23,C1              [1633]
440 DATA C1,C9,FE,C5,D0,D6,BA,D1              [751]
450 DATA EB,4F,06,00,21,1C,A6,09              [1413]
460 DATA C3,BB,DD,21,91,B2,75,C3              [1126]
470 DATA 68,A2,E5,3A,36,A6,32,80              [1430]
480 DATA BC,2A,37,A6,22,81,BC,CD              [1480]
490 DATA 80,BC,21,80,BC,36,C3,21              [1421]
500 DATA B2,A2,22,81,BC,E1,D8,C8              [1269]
510 DATA FE,1A,37,3F,C0,B7,37,C9              [1051]
520 DATA CD,D5,A5,F3,D9,CB,59,D9              [1082]
530 DATA FB,C0,E1,E3,CD,55,DD,30              [1455]
540 DATA 0A,3E,02,CD,FB,C1,E5,CD              [2199]
550 DATA 9F,BB,E1,E3,E9,CD,DB,A5              [1355]
560 DATA 18,03,CD,DE,A5,F3,D9,CB              [1527]
570 '*********************** Block 3          [1447]
580 DATA 18665                                [305]
590 DATA 59,D9,FB,C0,E1,E3,CD,55              [1531]
600 DATA DD,DC,0E,A3,E3,E9,CD,4B              [2240]
610 DATA C2,CD,D8,A5,F3,D9,CB,59              [1024]
620 DATA D9,FB,C0,CD,55,DD,D0,3E              [1468]
630 DATA 04,CD,FB,C1,E5,CD,59,BC              [1153]
640 DATA E1,C9,2B,7E,FE,20,28,FA              [1225]
650 DATA FE,2C,C0,2B,E5,2A,34,AE              [1763]
660 DATA CD,3F,DD,21,28,A6,CD,AA              [1804]
670 DATA FF,E1,38,08,FE,BB,C0,CD              [1770]
680 DATA 93,BB,18,03,CD,E1,BB,D1              [754]
690 DATA C3,4F,D0,3A,2F,A6,2F,32              [1266]
700 DATA 33,A6,CD,3C,18,AF,32,33              [714]
710 DATA A6,C9,3A,33,A6,B7,20,0F              [921]
720 DATA 3A,39,B3,B8,C2,CC,B1,3A              [605]
730 DATA 30,A6,B7,CA,CC,B1,C9,3A              [1876]
740 DATA 46,B3,B7,3A,2F,A6,20,11              [1515]

750 '*********************** Block 4          [1454]
760 DATA 15557                                [404]
770 DATA 07,32,2F,A6,3A,38,B3,47              [1267]
780 DATA DA,CC,B1,3A,39,B3,47,18              [1282]
790 DATA DE,E5,D5,C5,21,CF,B1,5F              [1766]
800 DATA 3A,07,B2,ED,44,47,AF,CB              [1472]
810 DATA 01,30,05,CB,03,30,01,B6              [2015]
820 DATA 23,10,F4,4F,7B,32,2F,A6              [960]
830 DATA 3A,30,A6,B7,20,0A,2A,38              [1225]
840 DATA B3,7D,AC,A1,AC,C1,18,01              [1524]
850 DATA F1,47,D1,E1,C3,CC,B1,CA              [1180]
860 DATA A3,FE,CD,86,0C,32,3A,B3              [1271]
870 DATA CD,1A,16,CD,FF,16,D2,68              [1264]
880 DATA A4,E5,FD,E1,D5,DD,E1,CD              [1183]
890 DATA A9,0B,3A,3A,B3,57,AE,A1              [1195]
900 DATA 28,7E,3A,38,B3,5F,AE,A1              [1188]
910 DATA 28,76,06,00,FD,23,CD,2D              [458]
920 DATA 0C,3A,34,B3,FD,95,38,52              [1326]

930 '*********************** Block 5          [1453]
940 DATA 15966                                [355]
950 DATA 7A,AE,A1,28,4D,7B,AE,A1              [2160]
960 DATA 20,EA,18,46,AE,77,CB,20              [1201]
970 DATA E5,CB,01,DC,05,0C,7A,AE              [1551]
980 DATA A1,28,12,7B,AE,A1,28,0D              [1077]
990 DATA CB,C0,CB,48,20,07,DD,2B              [1427]
1000 DATA DF,8A,A4,DD,23,CB,A0,CB             [1782]
1010 DATA 09,E1,E5,CB,09,DC,F9,0B             [984]
1020 DATA 7A,AE,A1,28,12,7B,AE,A1             [1767]
1030 DATA 28,0D,CB,E0,CB,68,20,07             [733]
1040 DATA DD,23,DF,8A,A4,DD,2B,CB             [1367]
1050 DATA 01,E1,3A,36,B3,FD,95,30             [1917]
1060 DATA 0F,FD,2B,CD,13,0C,7B,AE             [1894]
1070 DATA A1,28,05,7A,AE,A1,20,A4             [1286]
1080 DATA 2A,34,A6,7D,B4,C8,2B,22             [1518]
1090 DATA 34,A6,DF,78,A4,C3,D3,A3             [1234]
1100 DATA 7B,A4,FD,2A,89,AE,2B,56             [2087]

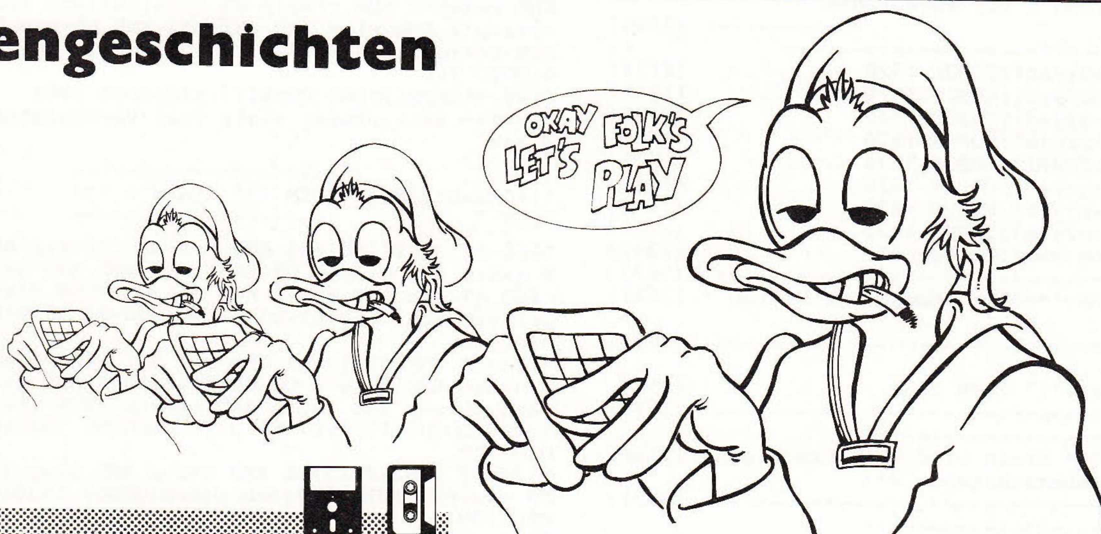
1110 '*********************** Block 6         [1460]
1120 DATA 17160                               [385]
1130 DATA 2B,5E,2B,22,89,AE,6E,26             [1822]
1140 DATA 00,C9,8D,A4,FD,C5,D5,2A             [466]
1150 DATA 89,AE,01,03,00,CD,18,F6             [1345]
1160 DATA 38,17,EB,22,89,AE,DD,E5             [847]
1170 DATA D1,FD,E5,C1,2B,72,2B,73             [1147]
1180 DATA 2B,71,2A,34,A6,23,22,34             [1282]
1190 DATA A6,D1,C1,C9,FE,03,28,41             [1373]
1200 DATA FE,AB,28,57,FE,9F,CA,4A             [1386]
1210 DATA A5,FE,2C,28,03,FE,28,C0             [1562]
1220 DATA 3E,02,B9,20,64,F1,C1,F5             [2185]
1230 DATA 3E,F8,B8,C0,79,FE,EE,28             [1224]
1240 DATA 15,FE,F5,C0,7E,2B,FE,72             [1151]
1250 DATA 20,FA,7E,3C,20,F6,23,CD             [1430]
1260 DATA 3F,DD,CD,37,DD,28,F1,CD             [999]
1270 DATA FB,CE,CD,37,DD,2C,C3,F5             [1746]
1280 DATA F8,3E,0D,BB,C0,2A,34,AE             [1472]

1290 '*********************** Block 7         [1459]
1300 DATA 19901                               [285]
1310 DATA CD,3F,DD,FE,7C,28,03,FE             [1396]
1320 DATA 83,C0,E1,2A,C2,B0,CD,21             [556]
1330 DATA FB,37,C9,3E,05,BB,C0,3E             [1681]
1340 DATA 1A,91,C0,B8,C0,F1,CD,F5             [1012]
1350 DATA C1,CD,D3,C1,CD,37,DD,29             [526]
1360 DATA CD,60,BB,E5,CD,19,FA,E1             [1119]
1370 DATA C9,3E,49,B9,C0,F1,C1,F5             [872]
1380 DATA 3E,D0,B8,C0,3E,93,B9,C0             [1553]
1390 DATA F1,E5,00,00,AF,CD,0A,FF             [1716]
1400 DATA E1,C9,F1,C1,F5,3E,C8,B8             [1410]
1410 DATA C0,3E,D6,B9,C0,3E,8B,BE             [998]
```

```
1420 DATA C0,F1,11,61,A5,1B,C3,DA         [846]
1430 DATA C8,06,00,FF,FF,C9,00,CD         [2264]
1440 DATA 55,DD,38,10,3E,02,CD,FB         [1856]
1450 DATA C1,B7,CC,84,BB,C4,81,BB         [393]
1460 DATA CD,55,DD,D0,3E,02,CD,FB         [1575]
1470 '************************ Block 8    [1434]
1480 DATA 19052                           [342]
1490 DATA C1,B7,CA,7E,BB,C3,7B,BB         [668]
1500 DATA 23,01,E4,BB,FE,BA,CA,18         [2158]
1510 DATA C2,FE,BB,C2,C6,DD,CD,55         [917]
1520 DATA DD,38,0A,CD,4B,C2,CD,D8         [599]
1530 DATA A5,CD,55,DD,D0,3E,02,CD         [2270]
1540 DATA FB,C1,32,30,A6,C9,CD,4B         [889]
1550 DATA C2,E5,DF,C7,A3,E1,C9,CD         [950]
1560 DATA 55,DD,38,0A,CD,67,CE,32         [1889]
1570 DATA 2F,A6,CD,55,DD,D0,3E,02         [1892]
1580 DATA CD,FB,C1,32,31,A6,C9,CD         [2316]
1590 DATA 09,BB,38,FB,C9,CF,A9,92         [1262]
1600 DATA CF,F6,97,CF,F4,95,CF,F1         [723]
1610 DATA 95,46,49,4C,CC,DD,47,52         [1078]
1620 DATA 41,50,48,49,43,D3,DE,4D         [895]
1630 DATA 41,53,CB,DF,46,52,41,4D         [1794]
1640 DATA C5,E0,43,55,52,53,4F,D2         [1101]

1650 '************************ Block 9    [1433]
1660 DATA 6328                            [308]
1670 DATA E1,43,4C,45,41,52,20,49         [1288]
1680 DATA 4E,50,55,D4,E2,43,4F,50         [1622]
1690 DATA 59,43,48,52,A4,7E,44,45         [1573]
1700 DATA 52,D2,49,00,AE,A5,88,A5         [952]
1710 DATA B7,A5,19,BD,67,A5,CF,A5         [1232]
1720 DATA 94,95,AE,AF,BC,BD,00,FF         [1309]
1730 DATA 00,FF,FF,00,00,00,00,00         [1953]
1740 DATA 00,EOF,UFF!                     [290]
```

Entengeschichten



Daß man auch mit 58 Jahren noch in der Lage sein kann, anspruchsvolle Computerspiele zu schreiben und Homecomputer nicht als neumodischen "Technik-Kram" abzutuen, beweist unser Leser R. Bodinger, der uns mit seinen "Entengeschichten" ein Geduldsspiel schickte, neben dem Rubicks Zauberwürfel aussieht wie ein Intelligenztest für Zweijährige.

Ihre Aufgabe ist es, die Ente über ein Spielfeld mit unregelmäßigen Steinen nach Hause zu führen. Diese unregelmäßigen Steine können verschoben werden.

Das Programm ist zwar leicht zu bedienen - eine genaue Anleitung ist im Programm enthalten - hat aber einen so verzwickten Lösungsweg, daß wir den Autor baten, bitte noch eine Liste der erforderlichen Züge mit einzugeben. Außerdem hat es einen Demo-Mode, in dem es sich selbst löst.

Wir hoffen, unsere Leser damit vor unnötigen Kosten zu bewahren, indem sie ihren Rechner irgendwann frustriert in die Ecke werfen.

Viel Spaß beim Tüfteln!!

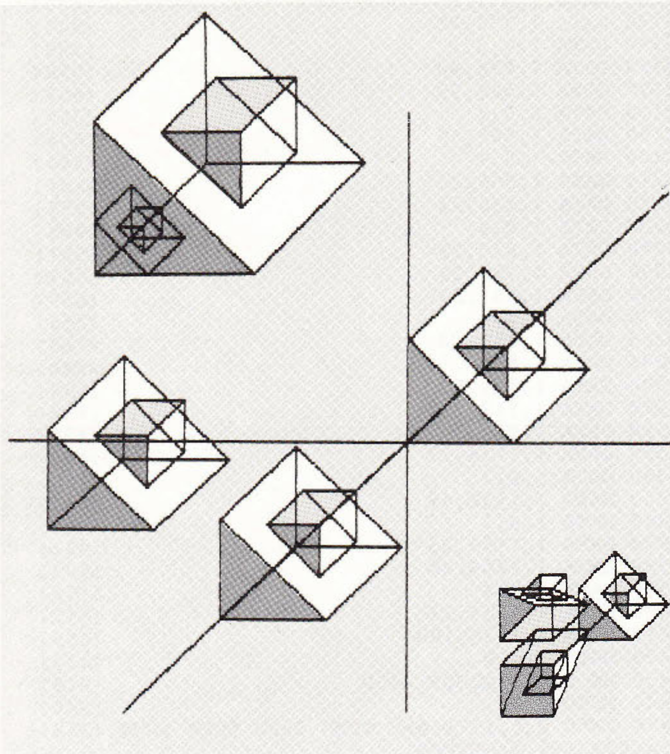
(Rudolf Bodinger/TM)

```
10 '    ************************          [978]
20 '    *                      *          [175]
30 '    *    PUZZLE          a *
40 '    *                      *          [175]
50 '    * (c) by Rudolf Bodinger *        [661]
60 '    *                      *          [175]
70 '    ************************          [978]
80 '                                      [117]
90 FOR i=1 TO 2000:NEXT                   [998]
100 CLEAR:MODE 1:BORDER 10:INK 0,0:INK 0,1 [1505]
110 '-------------------------------      [1400]
120 REM *** Bildschirmumrahmung ***       [1279]
130 '-------------------------------      [1400]
140 w=639:w1=399                          [775]
150 CLS: TAG:ORIGIN 1,1:FOR i=1 TO 10 STEP [1482]
 3
160 MOVE i,i                              [392]
170 DRAWR 0,w1-i*2,1:DRAWR w-i*2,0,1:DRAWR [4582]
 0,-w1+(i*2),1:DRAWR -w+(i*2),0,1
180 NEXT                                  [350]
190 FOR i=140 TO 150 STEP 3               [761]
200 MOVE i,i:DRAWR 0,w1-i*2,1:DRAWR w-i*2, [5700]
0,1:DRAWR 0,-w1+(i*2),1:DRAWR -w+(i*2),0,1
:NEXT
210 MOVE 233,225:PRINT"P U Z Z L E";     [2203]
220 MOVE 232,190:PRINT"Donald Duck";     [3105]
230 FOR i=1 TO 2000:NEXT                  [998]
240 FOR i=140 TO 200:MOVE i,i             [1193]
250 DRAWR 0,w1-i*2,0:DRAWR w-i*2,0,0:DRAWR [3800]
 0,-w1+(i*2),0:DRAWR -w+(i*2),0,0
260 NEXT                                  [350]
270 MOVE 33,310:DRAWR 0,0,1:PRINT"Das Prog [5883]
ramm bietet die Moeglichkeit";
280 MOVE 70,280:PRINT"das Puzzle automatis [3492]
ch ablaufen";
290 MOVE 126,250:PRINT"zu lassen um die Lo [3055]
esung";
300 MOVE 40,220:PRINT"zu sichten,oder den  [3727]
Code auszugeben";
310 MOVE 40,190:PRINT"falls Sie die Loesun [3718]
g nicht finden.";
320 MOVE 137,130:PRINT"Auto=1,Code=2,Manue [3159]
l=3";
330 MOVE 217,100:PRINT"Ihre Eingabe!";   [2439]
340 z$=INKEY$:IF z$=""OR z$>"3"THEN 340   [2016]
350 ON VAL(z$) GOTO 360,2470,380          [1451]
360 MOVE 60,70:PRINT"Speed 0-9? (0=schnell [3954]
,9=langsam)";
370 u$=INKEY$:IF u$=""OR u$>"9"THEN 370   [1290]
380 FOR i=55 TO 326:MOVE 33,i:DRAW 610,i,0 [3105]
:NEXT
390 ORIGIN 73,80                          [243]
400 '------------------------------------- [2997]
----------------------
410 REM   *** Variablen fuer die Zeichnung [3032]
 der Steine. k=pen ***
420 '------------------------------------- [2997]
----------------------
430 p=52:p1=50:p2=102:p3=19:p4=45:p5=30:p6 [2478]
=59:k=1
440 '------------------------------------- [3588]
-----------------------------------
450 REM *** Koordinat. a-r1 fuer Position. [4921]
```

```
der Steine (q-r1=freie Plaetze) ***
460 '---------------------------------------- [3588]
------------------------------
470 a=1+p:a1=1+p*3:b=1+p:b1=1+p*2:c=1:c1=1 [3657]
+p*3:d=1:d1=1+p
480 e=1+p*3:e1=1+p*3:f=1+p*3:f1=1+p:g=1+p: [3727]
g1=1+p:h=1+p*2
490 h1=1+p:m=1+p:m1=1:n=1+p*2:n1=1:q=1:q1= [3119]
1:r=1+p*3:r1=1
500 '---------------------------------------- [2989]
-----------------
510 REM *** Ausgangsposition des Spieles w [2655]
ird gezeichnet ***
520 '---------------------------------------- [2989]
-----------------
530 '                        ------------ [2366]
------------------------------------
540 x$="0":x=a:y=a1:GOSUB 1420:'*** Die Ko [4465]
ordinaten werden x u.y zugeordnet ***
550 '                        ------------ [2366]
------------------------------------
560 x$="1":x=b:y=b1:GOSUB 1520 [2154]
570 x$="2":x=c:y=c1:GOSUB 1570 [1261]
580 x$="3":x=d:y=d1:GOSUB 1570 [2456]
590 x$="4":x=e:y=e1:GOSUB 1570 [705]
600 x$="5":x=f:y=f1:GOSUB 1570 [1005]
610 x$="6":x=g:y=g1:GOSUB 1620 [1075]
620 x$="7":x=h:y=h1:GOSUB 1620 [1595]
630 x$="8":x=m:y=m1:GOSUB 1620 [2238]
640 x$="9":x=n:y=n1:GOSUB 1620 [2302]
650 '---------------------------------------- [1873]
660 GOSUB 1670:'*** Umrahmung der Steine * [2574]
**
670 '---------------------------------------- [1873]
-
680 k=0: IF z$="1" THEN 2580 [553]
690 ' ----------------------------------- [3171]
------------------------------
700 k=0:REM *** Stein wird bei Verschiebun [3760]
g mit pen 0 ueberschrieben ***
710 ' ----------------------------------- [3171]
------------------------------
720 IF a=1+p AND a1=1 THEN 1960 [966]
730 IF z$="1" THEN 2620 [1317]
740 MOVE 250,240:PRINT "Fuehren Sie Donald [2248]
";
750 MOVE 280,200:PRINT"Duck nach Haus!"; [3462]
760 MOVE 245,80:PRINT"Neubeginn=Leertaste" [3705]
;
770 MOVE 245,60:PRINT"Neuer Start=Taste n" [2900]
;
780 MOVE -45,-50:PRINT"Eingabe 0-9 + eine [4702]
der Tasten ";CHR$(240);" ";CHR$(241);" ";C
HR$(242);" ";CHR$(243);
790 MOVE 1+p+p3,-30:PRINT " ";:MOVE 1+p*2+ [2536]
p3,-30:PRINT " ";
800 x$=INKEY$:IF x$=""THEN 800 [2668]
810 IF x$=" "THEN 1930 [1163]
820 IF x$="n"THEN 10 [1009]
830 IF x$>"9"THEN 890 [901]
840 MOVE 1+p+p3,-30:PRINT x$; [1324]
850 p$=INKEY$:IF p$=""THEN 850 [1489]
860 IF p$=CHR$(240) OR p$=CHR$(241) OR p$= [4417]
CHR$(242) OR p$=CHR$(243) THEN 910
870 IF p$=" "THEN 1930 [764]
880 '                     ----------------- [2925]
-------------------------------------
890 GOSUB 1920:GOTO 700: REM *** Falsche E [5052]
ingabe. Tastenabfrage wiederholen ***
900 '                     ----------------- [2925]
-------------------------------------
910 MOVE 1+p*2+p3,-30:PRINT p$; [1244]
920 MOVE 64,280:PRINT" ";x$;" ";p$;" "; [1031]
930 IF x$="0"THEN 970 [1487]
940 ON VAL(x$) GOTO 1140,1260,1280,1300,13 [3221]
20,1340,1360,1380,1400
950 GOTO 700 [425]
960 '          ------------------------ [1287]
970 x=a:y=a1:    REM *** Stein 0 rauf *** [1454]
980 '          ------------------------ [1287]
990 IF CHR$(240)=p$ AND r=x AND r1=y+p*2 A [10810]
ND q=x+p AND q1=y+p*2 OR CHR$(240)=p$ AND
q=x AND q1=y+p*2 AND r=x+p AND r1=y+p*2 TH
EN q1=y:r1=y:GOSUB 1420:a1=a1+p:y=a1:GOSUB
 1420:GOTO 700:          REM *** Stein rauf
***
1000 '-------------------------- [1409]
1010 REM *** Stein 0 runter *** [1091]
1020 '-------------------------- [1409]
1030 IF CHR$(241)=p$ AND r=x AND r1=y-p AN [7624]
D q=x+p AND q1=y-p OR CHR$(241)=p$ AND q=x
 AND q1=y-p AND r=x+p AND r1=y-p THEN r1=y
+p:q1=y+p:GOSUB 1420:a1=a1-p:y=a1:GOSUB 14
20:GOTO 700
1040 '----------------------------- [1416]
1050 REM *** Stein 0 nach links *** [599]
1060 '----------------------------- [1416]
1070 IF CHR$(242)=p$ AND q=x-p AND q1=y AN [10553]
D r=x-p AND r1=y+p OR CHR$(242)=p$ AND r=x
-p AND r1=y AND q=x-p AND q1=y+p THEN r=x+
p:q=x+p:GOSUB 1420:a=a-p:x=a:GOSUB 1420:GO
TO 700
1080 '------------------------------- [1400]
1090 REM *** Stein 0 nach rechts *** [1440]
1100 '------------------------------- [1400]
1110 IF CHR$(243)=p$ AND q=x+p*2 AND q1=y [10076]
AND r=x+p*2 AND r1=y+p OR CHR$(243)=p$ AND
 r=x+p*2 AND r1=y AND q=x+p*2 AND q1=y+p T
HEN r=x:q=x:GOSUB 1420:a=a+p:x=a:GOSUB 142
0:GOTO 700
1120 TAGOFF:PRINT CHR$(7):TAG:GOTO 700: [6149]
REM *** Kein freier Platz fuer Verschiebun
g ***
1130 '          ------------------ [1051]
1140 x=b:y=b1:    REM *** Stein 1 *** [2099]
1150 '          ------------------ [1051]
1160 IF CHR$(240)=p$ AND r=x AND r1=y+p AN [9289]
D q=x+p AND q1=y+p OR CHR$(240)=p$ AND q=x
 AND q1=y+p AND r=x+p AND r1=y+p THEN q1=y
:r1=y:GOSUB 1520:b1=b1+p:y=b1:GOSUB 1520:G
OTO 700
1170 IF CHR$(241)=p$ AND r=x AND r1=y-p AN [6755]
D q=x+p AND q1=y-p OR CHR$(241)=p$ AND q=x
 AND q1=y-p AND r=x+p AND r1=y-p THEN r1=y
:q1=y:GOSUB 1520:b1=b1-p:y=b1:GOSUB 1520:G
OTO 700
1180 IF CHR$(242)=p$ AND q=x-p AND q1=y TH [4827]
EN q=x+p:GOSUB 1520:b=b-p:x=b:GOSUB 1520:G
OTO 700
1190 IF CHR$(242)=p$ AND r=x-p AND r1=y TH [4290]
EN r=x+p:GOSUB 1520:b=b-p:x=b:GOSUB 1520:G
OTO 700
1200 IF CHR$(243)=p$ AND q=x+p*2 AND q1=y [4536]
THEN q=x:GOSUB 1520:b=b+p:x=b:GOSUB 1520:G
OTO 700
1210 IF CHR$(243)=p$ AND r=x+p*2 AND r1=y [3606]
THEN r=x:GOSUB 1520:b=b+p:x=b:GOSUB 1520:G
OTO 700
1220 GOSUB 1920:GOTO 700 [1922]
1230 '--------------------------------------- [2862]
--------------------
1240 REM *** Zuordnung der Koordinaten fue [2331]
r die Steine 2-9 ***
1250 '--------------------------------------- [2862]
--------------------
1260 x=c:y=c1:GOSUB 1720 [1965]
1270 c=x:c1=y:GOTO 700 [1722]
1280 x=d:y=d1:GOSUB 1720 [849]
1290 d=x:d1=y:GOTO 700 [1997]
1300 x=e:y=e1:GOSUB 1720 [1496]
1310 e=x:e1=y:GOTO 700 [1975]
1320 x=f:y=f1:GOSUB 1720 [2095]
1330 f=x:f1=y:GOTO 700 [1176]
1340 x=g:y=g1:GOSUB 1820 [1540]
1350 g=x:g1=y:GOTO 700 [1103]
1360 x=h:y=h1:GOSUB 1820 [1701]
1370 h=x:h1=y:GOTO 700 [1441]
1380 x=m:y=m1:GOSUB 1820 [1097]
1390 m=x:m1=y:GOTO 700 [720]
1400 x=n:y=n1:GOSUB 1820 [1419]
1410 n=x:n1=y:GOTO 700 [969]
1420 '----------- [597]
1430 FOR i=1 TO 3 [435]
1440 MOVE x+i,y+i:DRAWR 0,(p2-i*2),k:DRAWR [6364]
 (p2-i*2),0,k:DRAWR 0,-(p2-i*2),k:DRAWR -(
p2-i*2),0,k:NEXT
1450 MOVE x+p4,y+p6:PRINT x$; [1332]
1460 MOVE x+22,y+30:PRINT"Duck"; [1052]
1470 '                      ---------- [1188]
------------
1480 IF x=1+p AND y=1 THEN 1960:REM Duck a [3886]
t home.Sound
1490 '                      ---------- [1188]
------------
1500 k=1:RETURN [771]
1510 '----------- [597]
```

```
1520 FOR i=1 TO 3                                  [435]
1530 MOVE x+i,y+i:DRAWR 0,(p1-i*2),k:DRAWR [4539]
 (p2-i*2),0,k:DRAWR 0,-(p1-i*2),k:DRAWR -(
p2-i*2),0,k:NEXT
1540 MOVE x+p4,y+p5:PRINT x$;                     [1338]
1550 k=1:RETURN                                    [771]
1560 '-----------                                  [597]
1570 FOR i=1 TO 3                                  [435]
1580 MOVE x+i,y+i:DRAWR 0,(p2-i*2),k:DRAWR [6428]
 (p1-i*2),0,k:DRAWR 0,-(p2-i*2),k:DRAWR -(
p1-i*2),0,k:NEXT
1590 MOVE x+p3,y+p6:PRINT x$;                     [2001]
1600  k=1:RETURN                                   [771]
1610 '-----------                                  [597]
1620 FOR i=1 TO 3                                  [435]
1630 MOVE x+i,y+i:DRAWR 0,(p1-i*2),k:DRAWR [4135]
 (p1-i*2),0,k:DRAWR 0,-(p1-i*2),k:DRAWR -(
p1-i*2),0,k:NEXT
1640 MOVE x+p3,y+p5:PRINT x$;                     [1196]
1650 k=1:RETURN                                    [771]
1660 '-----------                                  [597]
1670 FOR i=1 TO 22                                 [319]
1680 MOVE -2-i,-2-i:DRAWR 0,264+(i*2),k:DR [6338]
AWR 212+(i*2),0,k:DRAWR 0,-(264+i*2),k:DRA
WR -(212+i*2),0,k:NEXT
1690 MOVE 32,-6:PRINT" h o m e ";                 [2712]
1700 RETURN                                        [555]
1710 '----------------------                      [1174]
1720 REM *** Steine 2-5 ***                        [572]
1730 '----------------------                      [1174]
1740 IF CHR$(240)=p$ AND r=x AND r1=y+p*2  [6318]
THEN r1=y:GOSUB 1570:y=y+p:GOSUB 1570:RETU
RN
1750 IF CHR$(240)=p$ AND q=x AND q1=y+p*2  [3254]
THEN q1=y:GOSUB 1570:y=y+p:GOSUB 1570:RETU
RN
1760 IF CHR$(241)=p$ AND q=x AND q1=y-p TH [4604]
EN q1=y+p:GOSUB 1570:y=y-p:GOSUB 1570:RETU
RN
1770 IF CHR$(241)=p$ AND r=x AND r1=y-p TH [3655]
EN r1=y+p:GOSUB 1570:y=y-p:GOSUB 1570:RETU
RN
1780 IF CHR$(242)=p$ AND q=x-p AND q1=y AN [9669]
D r=x-p AND r1=y+p OR CHR$(242)=p$ AND r=x
-p AND r1=y AND q=x-p AND q1=y+p THEN r=x:
q=x:GOSUB 1570:x=x-p:GOSUB 1570:RETURN
1790 IF CHR$(243)=p$ AND r=x+p AND r1=y AN [6909]
D q1=y+p AND q=x+p OR CHR$(243)=p$ AND q=x
+p AND q1=y AND r1=y+p AND r=x+p THEN r=x:
q=x:GOSUB 1570:x=x+p:GOSUB 1570:RETURN
1800 GOSUB 1920:RETURN                            [1489]
1810 '----------------------                      [1174]
1820 REM *** Steine 6-9 ***                        [643]
1830 '----------------------                      [1174]
1840 IF CHR$(240)=p$ AND r=x AND r1=y+p TH [5059]
EN r1=y:GOSUB 1620:y=y+p:GOSUB 1620:RETURN
1850 IF CHR$(240)=p$ AND q=x AND q1=y+p TH [4868]
EN q1=y:GOSUB 1620:y=y+p:GOSUB 1620:RETURN
1860 IF CHR$(241)=p$ AND r=x AND r1=y-p TH [3910]
EN r1=y:GOSUB 1620:y=y-p:GOSUB 1620:RETURN
1870 IF CHR$(241)=p$ AND q=x AND q1=y-p TH [4111]
EN q1=y:GOSUB 1620:y=y-p:GOSUB 1620:RETURN
1880 IF CHR$(242)=p$ AND q=x-p AND q1=y TH [4814]
EN q=x:GOSUB 1620:x=x-p:GOSUB 1620:RETURN
1890 IF CHR$(242)=p$ AND r=x-p AND r1=y TH [4184]
EN r=x:GOSUB 1620:x=x-p:GOSUB 1620:RETURN
1900 IF CHR$(243)=p$ AND r=x+p AND r1=y TH [2919]
EN r=x:GOSUB 1620:x=x+p:GOSUB 1620:RETURN
1910 IF CHR$(243)=p$ AND q=x+p AND q1=y TH [5605]
EN q=x:GOSUB 1620:x=x+p:GOSUB 1620:RETURN
1920 TAGOFF:PRINT CHR$(7):TAG:RETURN              [3566]
1930 FOR i=1 TO 257                                [341]
1940 MOVE 1,1+i:DRAWR 208,0,0                     [1461]
1950 NEXT:k=1:GOTO 470                             [838]
1960 RESTORE 1970                                  [884]
1970 DATA 1,1136,23                                [454]
1980 DATA 1,0,4                                    [365]
1990 DATA 1,1136,48                                [441]
2000 DATA 1,1012,25                                [627]
2010 DATA 1,956,50                                 [476]
2020 DATA 1,758,25                                 [557]
2030 DATA 1,851,50                                 [674]
2040 DATA 1,1136,25                                [468]
2050 DATA 1,1012,50                                [614]
2060 DATA 1,568,23                                 [657]
2070 DATA 1,0,4                                    [365]
2080 DATA 1,568,46                                 [654]
2090 DATA 1,0,4                                    [365]
2100 DATA 1,568,23                                 [657]
2110 DATA 1,638,37                                 [565]
2120 DATA 1,758,13                                 [533]
2130 DATA 1,851,25                                 [679]
2140 DATA 1,758,75                                 [503]
2150 DATA 1,0,2                                    [359]
2160 DATA 1,758,48                                 [550]
2170 DATA 1,568,23                                 [657]
2180 DATA 1,0,4                                    [365]
2190 DATA 1,568,46                                 [654]
2200 DATA 1,0,4                                    [365]
2210 DATA 1,568,23                                 [657]
2220 DATA 1,638,23                                 [551]
2230 DATA 1,0,4                                    [365]
2240 DATA 1,638,23                                 [551]
2250 DATA 1,758,25                                 [557]
2260 DATA 1,851,23                                 [689]
2270 DATA 1,0,3                                    [362]
2280 DATA 1,851,12                                 [688]
2290 DATA 1,0,3                                    [362]
2300 DATA 1,851,23                                 [689]
2310 DATA 1,1136,25                                [468]
2320 DATA 1,956,50                                 [476]
2330 DATA 1,1136,23                                [454]
2340 DATA 1,0,3                                    [362]
2350 DATA 1,1136,48                                [441]
2360 DATA 1,0,3                                    [362]
2370 DATA 1,1136,23                                [454]
2380 DATA 1,1204,48                                [603]
2390 DATA 1,0,4                                    [365]
2400 DATA 1,1204,23                                [612]
2410 DATA 1,1136,100                               [559]
2420 DATA 0,0,0                                    [337]
2430 ON SQ(1) GOSUB 2450                           [918]
2440 GOTO 2430                                     [349]
2450 READ u,t,l:IF u=0 THEN 2460 ELSE SOUN [3697]
D u,t/4,l*1.5:RETURN
2460 GOTO 90                                       [312]
2470 TAGOFF:CLS:LOCATE 8,8:PRINT"Code auf  [5809]
Drucker ausgeben?"
2480 LOCATE 5,10:PRINT"Wenn ja,erst Drucke [3323]
r einschalten,"
2490 LOCATE 10,12:PRINT"dann Taste j druck [3811]
en.
2500 LOCATE 4,16:PRINT "Andere Taste nur B [4509]
ildschirmausgabe."
2510 u$=INKEY$:IF u$=""THEN 2510                  [1462]
2520 RESTORE 2630:CLS:PRINT:FOR w=1 TO 154 [3718]
:READ x$,p$:PRINT x$;" ";CHR$(VAL(p$));" "
;
2530 IF u$="j"THEN PRINT#8,x$;",";p$;" ";         [2362]
2540 NEXT                                          [350]
2550 LOCATE 4,20:PRINT"Neuer Start? Eine T [5388]
aste druecken!"
2560 CALL &BB06                                    [393]
2570 GOTO 10                                       [392]
2580 RESTORE 2630                                  [888]
2590 FOR w=1 TO 154:READ x$,p$:p$=CHR$(VAL [2697]
(p$))
2600 FOR s=1 TO VAL(u$)*100:NEXT                  [1843]
2610 GOTO 930                                      [530]
2620 NEXT w                                        [361]
2630 DATA 3,241,1,242,9,243,7,241,5,242,9, [2971]
240,7,243,5,241,1,243,1,243,6,240
2640 DATA 6,242,8,240,8,240,5,242,7,242,9, [3879]
241,1,241,8,243,8,243,6,243,6,243
2650 DATA 3,240,5,240,7,242,7,242,9,242,9, [3144]
242,1,241,8,241,8,242,4,241,4,241
2660 DATA 0,243,5,240,5,240,3,243,7,240,7, [3867]
240,9,242,9,240,1,242,1,242,8,241
2670 DATA 8,243,6,241,6,241,3,243,7,243,9, [3143]
240,1,240,6,242,6,242,8,242,8,242
2680 DATA 3,241,4,241,7,243,7,243,9,243,9, [2110]
243,1,240,6,240,8,242,3,242,4,242
2690 DATA 7,241,7,241,9,243,9,241,1,243,1, [4368]
243,6,240,6,243,8,240,8,240,3,242
2700 DATA 6,241,6,241,5,241,5,241,2,243,8, [3143]
240,8,240,3,240,3,240,6,242,6,240
2710 DATA 5,241,2,241,8,243,3,240,6,240,5, [1884]
242,2,241,2,241,6,243,3,241,8,242
2720 DATA 6,240,6,240,2,240,2,240,4,242,7, [3654]
242,9,241,1,241,0,241,6,243,6,243
2730 DATA 8,243,8,243,3,240,2,240,5,240,4, [2926]
240,7,242,7,242,9,242,9,242,1,241
2740 DATA 0,241,6,241,8,243,2,243,4,240,4, [3505]
240,0,242,6,241,6,241,8,241,8,241
2750 DATA 2,243,4,243,3,243,5,240,5,240,0, [3518]
242,8,242,6,240,1,240,9,243,9,243
2760 DATA 7,243,7,243,0,241,8,242,8,242,6, [2684]
242,6,242,1,240,9,240,7,243,0,243
```

3D-Grafik mit LOGO



Warum wurde die Programmiersprache LOGO gewählt? Diese Sprache wurde aus mehreren Gründen gewählt. Der zwingendste Grund war wohl, daß mit dem Joyce in der Grundausstattung nur mit Logo Grafikprogrammierung möglich ist. Logo bietet dazu aber sehr schöne Möglichkeiten, da entsprechende Befehlsprozeduren zur Verfügung stehen. Mit dieser Sprache lassen sich die Programme sehr gut strukturiert gestalten. Ein weiterer Vorteil dürfte darin liegen, daß die Programme auch auf die anderen Computer von Schneider ohne größere Probleme übertragen werden können, sofern genügend Arbeitsspeicher vorhanden ist (CPC 464 und 664).

Wie bedient man das Programm?

Als erstes muß man seinen Joyce anschalten und ihn mit der zweiten Seite der Originalkopie oder der entsprechenden Sicherheitskopie, die das CP/M-Betriebssystem enthält, versorgen. Wenn die richtige Diskette gewählt worden ist, bootet der Joyce automatisch das CP/M-System nach dem Einschieben der Diskette in das Laufwerk. Nach erfolgreicher Tat erscheint das Systemprombt und erwartet weitere Eingaben. Diese werden nun auch getätigt, indem das CP/M-Dienstprogramm LANGUAGE mit 'language 0' aufgerufen wird (das Symbol hinter LANGUAGE ist eine Null!), womit auf den amerikanischen Zeichensatz umgeschaltet wird. Dieser wird benötigt, um beim Eintippen oder eventuellen Korrigieren des Programms durch Betätigung der Tastenfolge <SHIFT>-ä, die eckige Klammer auf, und bei der Tastenfolge <SHIFT>-ü, die eckige Klammer zu, zu bekommen. Im deutschen Zeichensatz werden diese Tastenfolgen als 'Ä' und 'Ü' auf dem Bildschirm angezeigt, der Computer interpretiert diese aber richtig als die eckigen Klammern. Diese Verwirrung haben wir dem ASCII-Code zu verdanken (American Standard Code of Information Interchange).

Nachdem dieses geschehen ist, schieben wir Seite 4 der Systemdiskette oder deren Kopie in das Laufwerk und tippen LOGO, gefolgt von einem <RETURN>, ein. Der Computer lädt jetzt den Logo-Interpreter, was einige Sekunden dauert. Wenn Logo nun bereit ist, Ihre Befehle entgegenzunehmen, wird es Sie ein wenig Zeit und ein, zwei Tassen Kaffee kosten, bis das Programm eingetippt ist.

Wenn im Prozedurteil INSTALLIEREN sinnvolle Werte eingegeben wurden, so können Sie das Programm mit dem Aufruf der Prozedur HAUPTPROGRAMM starten. Die Prozedur benötigt keine Parameter zur Übergabe. Wenn Sie nun verärgert feststellen, daß das Programm nicht auf Anhieb läuft, so gehören Sie zu den 90 %, bei denen sich bei der Abtipperei Fehler eingeschlichen haben. Der Logo-Interpreter ist in diesen Dingen jedoch recht komfortabel, da er genau beschreibt, wo der Fehler liegt. Mit der Programmvorlage und den Fehlermeldungen des Interpreters sollte es Ihnen gelingen, das Programm zum Funktionieren zu bringen.

Nach Aufruf der Prozedur HAUPTPROGRAMM wird auf dem Bildschirm nun einiges geschehen. Der Textbildschirm ist auf die unteren drei Zeilen begrenzt, in welchem ein Menue ausgegeben wird. Das erste Menue wird Sie danach fragen, ob Sie die Figur zoomen, drehen, verschieben oder ein Koordinatenkreuz zeichnen wollen. Bei Eingabe des jeweils ersten Buchstaben – also 'z', 'v', 'd' oder 'k' (wobei Groß-/Kleinschreibung egal ist) und Abschluß durch <RETURN>, wird der entsprechende Befehl ausgeführt. Wählen Sie 'k' für Koordinatenkreuz aus, so wird dieses entsprechende mit seinen drei Raumachsen in die Zeichenebene eingefügt. Bei entsprechender Wahl für Zoomen, Drehen oder Verschieben, wird jeweils ein weiteres Menue aufgerufen. Diese Menues fragen jeweils nach entsprechenden Werten für die x-, y- und z-Komponenten.

Beim Zoomen kann das in INSTALLIEREN definierte Objekt in x-, y-, z-Richtung um den Ursprung vergrößert werden. Der neutrale Faktor beim Zoomen ist der Wert 1: Soll das Objekt in einer Richtung unverändert bleiben, so muß für diese Komponente dieser Wert eingegeben werden. Einer der besonderen Fälle des Zoomens ist die Eingabe von 1 1 1, welche das Objekt nicht verändert; 0 1 1 bedeutet einen Schnitt in der yz-Ebene, 1 0 1 in der xz-Ebene und 1 1 0 in der xy-Ebene.

Für das Verschieben gilt formell dasselbe. Das neutrale Element hierbei ist der Wert 0 – in allen anderen Fällen wird das Objekt entsprechend in x-, y-, z-Richtung verschoben. Wird also 0 0 0 eingegeben, so bleibt das Objekt unverändert.

So, nun kommen wir zum letzten, schönsten und langsamsten, da kompliziertesten Befehl: dem Drehen. Hier werden nun die Winkel für die Drehung um die x-, y- und z-Achsen erwartet. Das neutrale Element der Drehung ist der Winkel 0. Sonderfälle für die Drehung sind die Eingaben von 0 0 0 oder 360 360 360, die das Objekt unverändert lassen. Die DREHEN-Prozedur ist die zeitaufwendigste, da viele Berechnungen durchgeführt werden müssen.

Wenn nun eine der Aktionen fertig ausgeführt worden ist, erscheint sofort das nächste Menue. Dieses stellt uns wieder vor vier Entscheidungen. Fangen wir hier mit der einfachsten und familienfreundlichsten Auswahl an:

Wenn Sie das Programm nun schon mehrere Stunden in der "Mache" haben und Sie sich vor Begeisterung kaum losreißen können – dies zum Erhalt des Hausfriedens aber immer dringlicher wird, so sollten Sie ein 'a', gefolgt von <RETURN>, eingeben. Dieser Befehl bricht das Programm ab. Sollten Sie nicht von diesem Umstand bedrängt sein, so können Sie die restlichen Menue-Eingaben ausprobieren.

Die Eingabe von 'w' bemächtigt Sie, das Objekt weiter zu bearbeiten. Dabei wird die letzte Darstellung des Objekts zu Grunde gelegt, d.h., war das Objekt schon gedreht, gezoomt oder verschoben worden, so wird von dieser letzten Objektdarstellung aus weitergearbeitet. Hier kann auch entschieden werden, ob der Bildschirm gelöscht oder die alte(n) Darstellung(en) erhalten bleiben soll(en).

Bei der Eingabe von 'n', für Neuanfang, wird der Bildschirm gelöscht und das Objekt wieder neu installiert. Dieses ist zu empfehlen, wenn man sich bei der Veränderung des Objekts aus dem Bildschirm herausgearbeitet hat.
Als letztes in diesem Menue bleibt einem noch die Möglichkeit, sein geschaffenes Bild auf Diskette zu speichern, um es für die Nachwelt zu erhalten. Dieses wird durch die Eingabe eines 'b' realisiert. Nach Eingabe des Dateinamens ohne Dateityp wird es auf die Diskette im Laufwerk A: als Logo-Bild-Datei (PICture-File) abgespeichert. Diese können später unter Logo mit der Standard-Prozedur LOADPIC "NAME wieder eingeladen werden.
Als letztes bleibt noch zu erwähnen, wie man das erschaffene Bild auf den Drucker ausgeben kann. Da es für den Joyce keine mir bekannte Logo-Prozedur gibt, die das Bild auf den Drucker ausgibt, muß man sich mit folgendem behelfen: Bei gleichzeitiger Betätigung der Tastenkombination <EXTRA> und <PTR> wird vom Bildschirm eine Hardcopy erstellt. Diese Funktion gilt beim Joyce allgemein und ist nicht nur auf die Programmiersprache Logo begrenzt.

Was leistet das Programm?

Das Programm kann Objekte, die im Prozedurteil INSTALLIEREN festgelegt werden, auf dem Bildschirm als Drahtgittermodel darstellen. Die Darstellung erfolgt in der 45-Grad-Perspektive. Die dargestellten Objekte können in folgenden Variationen weiter bearbeitet werden:
Sie können, wie schon beschrieben, um die Raumachsen gedreht werden. Die Drehung kann variiert werden, z.B. nur um eine, zwei oder um alle drei Achsen gleichzeitig.
Die Objekte können in x-, y- und z-Richtung vergrößert (gezoomt) werden. Dieses kann jeweils auch nur in einer oder mehreren gewünschten Richtung(en) erfolgen.
Die Objekte können in Richtung aller drei Raumachsen verschoben werden.
Die Objekte können, jeweils von ihrer ursprünglichen oder letzten Darstellung aus, bearbeitet werden.
Der Bildschirm kann, je nach Wunsch, nach jeder Bearbeitung gelöscht oder stehengelassen werden.
Die entstandenen Bilder können auf Diskette gespeichert werden.
Das Bild kann auf den Drucker als Hardcopy ausgegeben werden.
Sämtliche Befehlssteuerungen können über Menue angewählt werden; bis auf die Hardcopy, für die die Tastenkombination <EXTRA> und <PTR> zuständig ist.

Anwendungen für das Programm?

Einige von Ihnen werden sich sicher Fragen über die Brauchbarkeit des Programms stellen. Um die Zweifel darüber etwas zu beseitigen, bringe ich Ihnen ein paar Beispiele:
Jeder, der sich mit Elektronik oder Metallbearbeitung befaßt, wird schon mit technischen Zeichnungen zu tun gehabt haben. Zu technischen Zeichnungen gehören oft Darstellungen der Werkstücke in verschiedenen Ansichten. Diese Darstellungen müssen konstruiert werden, und dazu braucht man einiges an Übung und Vorstellungsvermögen, um sich z.B. Werkstücke, die man von vorne betrachtet, von der Seite vorzustellen. Hier könnte das Programm zur Arbeitserleichterung genutzt werden, um z.B. kleinere Werkstücke zu drehen und dann zu betrachten.
Ein weiteres Einsatzgebiet wird für jeden Schüler interessant sein, der sich irgendwann im Verlauf seines Schüler- oder Studentenlebens mit linearer Algebra, also mit Vektorrechnung, beschäftigen muß. Bei der Darstellung von Vektoren befindet man sich oft im dreidimensionalen Raum, so daß dieses Programm als Darstellungsmedium für Vektoren genutzt werden kann.
Drittens kann man sich dieses Programm als primitives CAD-System vorstellen und sich daraus einige Dinge für eigene Projekte abschauen.
Auch möchte ich darauf hinweisen, daß die Programmierung von Grafik nach meiner Meinung eines der interessantesten Gebiete ist, und es sich schon deshalb lohnt, sich damit zu beschäftigen.
Dies sind nur einige Anregungen dafür, wie das Programm verwendet werden kann. Es lassen sich bestimmt auch noch andere Gebiete finden, bei denen das Programm zu gebrauchen ist.

Die mathematischen Grundlagen

Hier sollen im Ungefähren die mathematischen Grundlagen für das Programm besprochen werden. Diese Beschreibung erhebt keinen Anspruch auf Vollständigkeit oder Fehlerlosigkeit. Es kann in dieser Beschreibung auch nur ein kleiner Abriß dessen erfolgen, was an Aufwand hinter diesem Problem steckt.
Ein Vektor im dreidimensionalen Raum wird mit der Angabe von drei Größen genau beschrieben. Diese Größen sind die x-, y- und z-Komponenten des Vektors. Schriftlich dargestellt wird dieser in runden Klammern, in denen die Komponenten untereinander aufgeführt sind: die x-Komponente als oberste Zahl, die y-Komponente als mittlere und die z-Komponente als letzte. So bedeutet z.B.:

1
2
3

daß man von irgendeinem Punkt im Raum, dem Anfangspunkt des Vektors, eine Einheit nach rechts gehen muß (x-), zwei Einheiten nach oben (y-) und drei Einheiten nach hinten (z-), um den Endpunkt zu erreichen. Wenn man sich von dem Anfangspunkt zu dem Endpunkt einen Pfeil vorstellt, so haben Sie die zeichnerische Darstellung des Vektors. Der Anfangspunkt sämtlicher Vektoren bezieht sich auf den Ursprung des Koordinatensystems.
Die Koordinaten der Punkte, die wir im Prozedurteil eingegeben haben, werden nun durch Vektoren dargestellt, die im Koordinatenursprung entstehen und auf diesen Punkt zulaufen. Diese Vektoren lassen sich nun in folgender Weise mit den Prozeduren des Programms verändern:
Wird die Prozedur ZOOMEN z.B. mit den Faktoren 2, 1 und 3 für die x-, y- und z-Komponenten aufgerufen, und sind die Punkte z.B. als folgende Vektoren im Programm enthalten,

10	30	50	20	30	20
20	40	10	10	20	80
50	40	30	40	10	0

so werden die x-Werte der Vektoren mit 2, die der y-Werte mit 1, die z-Werte mit 3 multipliziert und wir erhalten:

20	60	100	40	60	40
20	40	10	10	20	80
150	120	90	120	30	0

Wenn Sie das Ergebnis auf dem Bildschirm betrachten, so werden Sie feststellen, daß das Objekt sich in der Breite verdoppelt hat und doppelt so weit in der x-Richtung vom Ursprung entfernt ist. Die Höhe ist geblieben, die Tiefe hat sich verdreifacht und das Objekt hat sich in der z-Richtung um das Dreifache verschoben. Dieses kommt daher, daß die x-Komponenten je verdoppelt, die y-Komponenten unverändert blieben und die z-Komponenten verdreifacht wurden.

Bei Aufruf der Prozedur VERSCHIEBEN mit den Werten 12, 34 und 50 werden die Vektoren:

10	30	50	20	30	70
20	40	10	10	20	80
50	40	30	40	10	0

wie folgt verändert. Zu den x-Komponenten wird 10 addiert, zu den y-Komponenten 34, und zu den z-Komponenten 50. Es ergeben sich folgende Vektoren:

22	32	62	32	32	82
54	74	44	44	54	114
100	90	80	90	60	50

Bei Betrachtung des Bildschirms ergibt sich folgende Veränderung:

Das Objekt hat sich um 12 Einheiten nach rechts, um 34 Einheiten nach oben und um 50 Einheiten nach hinten verschoben.

Nun kommen wir zum mathematischen Prunkstück des Programms, der Prozedur DREHEN. Diese Prozedur kann die Vektoren um die x-, y- und z-Achse drehen. Um diese Prozedur mathematisch zu erklären, müssen wir noch über den Begriff der Matrix Bescheid wissen. Eine Matrix ist sozusagen eine Zusammenfassung mehrerer Vektoren. Bei dieser Prozedur gibt es nun gleich zwei von diesen Matrizen. Die erste Matrix entsteht, wenn man die ganzen x-Komponenten der Punktvektoren untereinander schreibt. Daneben schreibt man die ganzen y-Komponenten untereinander und daneben dann die ganzen z-Komponenten. Die Matrizenschreibweise ist praktisch nur eine besondere Darstellungsart für mehrere Vektoren. Matrizen werden wie Vektoren in runden Klammern dargestellt. So erhalten wir für unser bisheriges Beispiel folgende Schreibweise:

10	20	50
30	40	40
50	10	30
20	10	40
30	20	10
20	80	0

Jetzt benötigen wir für unsere DREHEN-Prozedur noch eine zweite Matrix. Diese Matrix enthält Werte, deren Berechnung in den tiefsten Dschungel der linearen Algebra führen würde. Die Herleitung dieser Berechnungen würde den Rahmen dieses Beitrages sprengen. Also müssen Sie die folgenden Tatsachen als gegeben hinnehmen. Die Prozedur DREHEN benötigt alle drei Drehwinkel um die jeweiligen Achsen, die dann lauten:

x-Winkel, um den das Objekt um die X-Achse gedreht wird.

y-Winkel, um den das Objekt um die Y-Achse gedreht wird.

z-Winkel, um den das Objekt um die Z-Achse gedreht wird.

Wenn diese Voraussetzungen erfüllt sind, erhalten wir folgende Drehmatrix:

cos(x)*cos(z)-sin(x)*sin(y)*sin(z)	sin(x)*cos(y)	cos(x)*sin(z)+sin(x)*sin(y)*cos(z)
-sin(x)*cos(z)-cos(x)*sin(y)*sin(z)	cos(x)*cos(y)	-sin(x)*sin(z)+cos(x)*sin(y)*cos(z)
-cos(y)*sin(z)	-sin(y)	cos(y)*cos(z)

Wenn wir nun die Winkel angeben, können wir die Drehmatrix erstmal ausrechnen. Als Drehwinkel sei z.B. für x 90 Grad, für y 0 Grad und für z 90 Grad. Das Ergebnis sieht in Matrixform nun so aus:

0	1	0
0	0	-1
-1	0	0

Die letzte Aktion, die uns nun noch bevorsteht, ist die Multiplikation beider Matrizen. Zur Durchführung dieser Matrizenmultiplikation schreiben wir die Matrizen in folgender Weise hin:

			0	1	0
			0	0	-1
			-1	0	0
10	20	50	e01	e02	e03
30	40	50	e04	e05	e06
50	10	30	e07	e08	e09
20	10	40	e10	e11	e12
30	20	10	e13	e14	e15
20	80	0	e16	e17	e18

Die Werte, die in der Mitte zwischen den beiden Matrizen stehen, sind die Ergebnisse e01 – e18. Diese werden folgendermaßen berechnet:

Steht das Ergebnis e in der i-ten Zeile und in der k-ten Spalte, so muß man das erste Element der k-ten Spalte der Drehmatrix mit dem ersten Element der i-ten Zeile der Wertematrix multiplizieren. Dieser Vorgang wiederholt sich mit dem zweiten Element, und dann mit dem dritten Element. Alle drei Werte müssen nun nur noch addiert werden, um das Ergebnis zu erhalten. So ergeben sich z.B. folgende Werte für:

e01=0*10+0*20+(-1)*50=-50
e08=1*50+0*10+0*30 = 50
e15=0*30+(-1)*20+0*10=-20

und wir erhalten folgende Ergebnismatrix:

-50	10	-20
-40	30	-40
-30	50	-10
-40	20	-10
-10	30	-20
0	20	-80

Aus dieser Ergebnismatrix lassen sich die Vektoren mit den gedrehten Koordinaten herauslesen, die dann lauten:

-50	-40	-30	-40	-10	0
10	30	50	20	30	20
-20	-40	-10	-10	-20	-80

Mit Beendigung dieses Schrittes ist die Prozedur DREHEN abgeschlossen.

Einige Anmerkungen zum Logo-Editor

Wenn während der Programmausführung ein Fehler entdeckt wird, muß er korrigiert werden. Dies geschieht, indem wir die fehlerhafte Prozedur mit **ed "Prozedurname** zum Editieren aufrufen. Nach diesem Aufruf befindet man sich im Edit-Modus und die Prozedur kann nun verbessert werden. Leider gelten bei Logo im Edit-Modus nicht die gewohnten Steuertasten, wie z.B. die Cursortasten – sie müssen über andere Tasten bedient werden. Die wichtigsten Bedienungselemente, die zum Editieren gebraucht werden, sind:

<ALT> F
Cursor um eine Stelle nach rechts bewegen.
<ALT> B
Cursor um eine Stelle nach links bewegen.
<ALT> P
Cursor um eine Stelle nach oben bewegen.
<ALT> N
Cursor um eine Stelle nach unten bewegen.
<ALT> A
Cursor zum Zeilenanfang bewegen.
<ALT> E
Cursor zum Zeilenende bewegen.
<ALT> H
links von Cursor löschen.
<ALT> D
rechts von Cursor löschen.
<ALT> K
von Cursor bis zum Zeilenende löschen.
<ALT> R
setzt den Cursor an die linke obere Anfangsposition.
<ALT> O
bringt den Rest der Zeile ab den Cursor in die nächste Zeile.
<ALT> C
beendet das Editieren und kehrt in den normalen Modus zurück.
<ALT> G
bricht das Editieren ab, ohne die gemachten Änderungen zu berücksichtigen.

Erläuterungen zum Programmteil HAUPTPROGRAM:

10HAUPTPROGRAM ruft folgende Prozeduren auf:
INSTALLIEREN
MENUE1
MENUE2

HAUPTPROGRAM wird von folgenden Prozeduren aufgerufen:
MENUE2

Verwendete Variablen sind: keine

Zeile 1
Prozeduranfang, Namensbezeichnung der Prozedur
Zeile 2
Einstellung des Bildschirms auf 3 Textzeilen
Zeile 3
Textbildschirm löschen
Zeile 4
Grafikbildschirm löschen
Zeile 5
Prozedur INSTALLIEREN aufrufen
Zeile 6
Prozedur MENUE1 aufrufen
Zeile 7
Prozedur MENUE2 aufrufen
Zeile 8
Prozedurende

```
to hauptprogram
setsplit 3
ct
cs
installieren
menue1
menue2
end
```

Erläuterungen zum Programmteil INSTALLIEREN:

INSTALLIEREN ruft folgende Prozeduren auf:
keine
INSTALLIEREN wird von folgenden Prozeduren aufgerufen:
HAUPTPROGRAM

Verwendete Variablen sind:
xw
Speicherung der X-Werte der eingegebenen Punkte
yw
Speicherung der Y-Werte der eingegebenen Punkte
zw
Speicherung der Z-Werte der eingegebenen Punkte
indvek
Speicherung der Zuordnungsvorschrift (in welcher Reihenfolge die Punkte verbunden werden sollen)
zm
Speicherung des Zeichenmodus, mit dem die Punkte verbunden werden sollen.
Zeile 1
Prozeduranfang, Namensbezeichnung der Prozedur
Zeile 2
In xw werden alle X-Werte der eingegebenen Punkte gespeichert. Z.B. bedeutet (10 20 30 10): X-Wert des ersten Punktes ist 10, des zweiten Punktes 20 usw.
Zeile 3
In yw werden alle Y-Werte der eingegebenen Punkte gespeichert. (20 40 30 20) bedeutet: Y-Wert des ersten Punktes ist 20, des zweiten Punktes 40, usw.
Zeile 4
In zw werden alle Z-Werte der eingegebenen Punkte gespeichert, wie Zeile 2 und 3.
Zeile 5
In indvek wird die Zuordnungsvorschrift gespeichert, die angibt, in welcher Reihenfolge die Punkte verbunden werden. So bedeutet z.B.(1 2 3 4 1 5), daß eine Verbindungslinie gezogen wird von dem Punkt, der in xw,yw,zw an erster Stelle steht, zu dem Punkt, der in xw,yw,zw an zweiter Stelle steht usw.
Zeile 6
In zm wird der Zeichenmodus abgespeichert, mit dem die Verbindungslinien gezeichnet werden sollen. So bedeutet z.B. (0 1 1 0 1) in Bezug auf Zeile 4, daß die Verbindungslinie von Punkt 1 nach Punkt 2 unsichtbar gezeichnet werden soll, die Verbindung von 2 nach 3 sichtbar, von 3 nach 4 sichtbar, von 4 nach 1 unsichtbar, usw. Es bedeuten also:
1 sichtbare Verbindungslinien
0 unsichtbare Verbindungslinien
Die Liste von zm ist um eins kürzer als die von indvek.
Zeile 7
Prozedurende

Allgemein:
Zwingend ist, daß das 1-te Element von xw,yw,zw 0 ist und, daß das 1-te Element von indvek 1 ist, da vom Ursprung aus begonnen wird.

```
to installieren
make "xw [0 10 50 50 10 10 50 50 10]
make "yw [0 10 10 20 20 10 10 20 20]
make "zw [0 0 0 0 0 80 80 80 80]
make "indvek [1 2 3 4 5 2 6 7 8 9 6 9 5 4 8 7 3]
make "zm [0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1]
end
```

Erläuterungen zum Programmteil MENUE1:

MENUE1 ruft folgende Prozeduren auf:
MENUE1ZOOMEN
MENUE1DREHEN
MENUE1VERSCHIEBEN
KOORDINATENKREUZ

MENUE1 wird von folgenden Prozeduren aufgerufen:
HAUPTPROGRAM
MENUE2
MENUE1
KOORDINATENKREUZ

Verwendete Variablen sind:
abfrage
dient zur Abfrage der Tastatureingabe

Zeile 1
Prozeduranfang, Namensbezeichnung der Prozedur
Zeile 2 + 3
Ausgabe von Menuetext auf Bildschirm
Zeile 4
In die Variable wird über Tastatur ein Zeichen eingegeben
Zeile 5
Abfrage, ob eingegebenes Zeichen gleich 'z' oder 'Z' ist. Wenn ja, so wird die Prozedur MENUE1-ZOOMEN aufgerufen
Zeile 6
Eingegebenes Zeichen gleich 'd' oder 'D'? Wenn ja, wird die Prozedur MENUE1DREHEN aufgerufen
Zeile 7
MENUE1VERSCHIEBEN aufrufen?
Zeile 8
KOORDINATENKREUZ aufrufen?
Zeile 9
Prozedurende

```
to menue1
pr [Figur ist instaliert]
pr [Figur [Z] oomen, [D] rehen, [V] erschieben,oder
 [K] oordinatenkreuz zeichnen]
make "abfrage rq
if or (:abfrage = "Z) (:abfrage = "z) [menue1zoomen]
if or (:abfrage = "D) (:abfrage = "d) [menue1drehen]
if or (:abfrage = "V) (:abfrage = "v) [menue1verschieben]
if or (:abfrage = "K) (:abfrage = "k) [koordinatenkreuz]
end
```

Erläuterungen zum Programmteil MENUE2:

MENUE2 ruft folgende Prozedur auf:
MENUE1
MENUE2
MENUE2MODUS
MENUE2SPEICHERN
HAUPTPROGRAM

MENUE2 wird von folgenden Prozeduren aufgerufen:
HAUPTPROGRAM
MENUE2
Verwendete Variablen sind:
abfrage
dient zur Abfrage der Tastatureingabe

Zeile 1
Prozeduranfang, Namensbezeichnung der Prozedur
Zeile 2 + 3
Ausgabe von Menuetext auf den Bildschirm
Zeile 4
In die Variable wird über die Tastatur eingegeben
Zeile 5
Objekt weiter bearbeiten? → MENUE2MODUS und dann MENUE1 aufrufen
Zeile 6
Neu anfangen? → HAUPTPROGRAMM
Zeile 7
Abbrechen?
Zeile 8
Bild speichern? → MENUE2SPEICHERN aufrufen
Zeile 9
Falls ein falsches Zeichen eingegeben worden ist, so ruft sich MENUE2 noch einmal selber auf
Zeile 10
Prozedurende

```
to menue2
pr [Figur [w] eiter bearbeiten, [N] euanfang, [A] brechen,
 [B] ild speichern]
pr []
make "abfrage rq
if or (:abfrage = "w) (:abfrage = "W) [menue2modus menue1]
if or (:abfrage = "n) (:abfrage = "N) [hauptprogram]
if or (:abfrage = "a) (:abfrage = "A) [stop]
if or (:abfrage = "b) (:abfrage = "B) [menue2speichern]
menue2
end
```

Erläuterungen zum Programmteil MENUE1 ZOOMEN:

MENUE1ZOOMEN ruft folgende Prozeduren auf:
ZOOMEN

MENUE1ZOOMEN wird von folgenden Prozeduren aufgerufen:
MENUE1

Verwendete Variablen sind:
eingabe
Tastaturabfrage. Liest eine Liste von drei Werten über die Tastatur ein
zoomx
Enthält den Zoomfaktor, um den die Figur in X-Richtung vergrößert werden soll
zoomy
Enthält den Zoomfaktor, um den die Figur in Y-Richtung vergrößert werden soll
zoomz
Enthält den Zoomfaktor, um den die Figur in Z-Richtung vergrößert werden soll

Zeile 1
Prozeduranfang, Namensbezeichnung der Prozedur
Zeile 2 + 3
Ausgabe von Menuetext auf den Bildschirm
Zeile 4
In **eingabe** wird eine Liste von drei Werten eingelesen. Die Eingabe muß mit <RETURN> abgeschlossen werden
Zeile 5
Der Variable zoomx wird das erste Element der Eingabeliste zugewiesen
Zeile 6
Der Variable zoomy wird das zweite Element der Eingabeliste zugewiesen

Zeile 7
Der Variable zoomz wird das dritte Element der Eingabeliste zugewiesen
Zeile 8
Prozedur ZOOMEN aufrufen
Zeile 9
Prozedurende

```
to menuelzoomen
pr [x,y,z Werte zum zoomen eingeben. Z.b 2 2 3
 < return >]
pr [Die Eingabe von 1 1 1 ergibt kein Zoomen]
make "eingabe rl
make "zoomx item 1 :eingabe
make "zoomy item 2 :eingabe
make "zoomz item 3 :eingabe
zoomen
end
```

Erläuterungen zum Programmteil MENUE1VERSCHIEBEN:

MENUE1VERSCHIEBEN ruft folgende Prozeduren auf:
VERSCHIEBEN

MENUE1VERSCHIEBEN wird von folgenden Prozeduren aufgerufen:
MENUE1

Verwendete Variablen sind:
eingabe
Tastaturabfrage. Liest eine Liste von drei Werten über die Tastatur ein
mx
Enthält den Verschiebungsfaktor, um den die Figur in X-Richtung verschoben werden soll
my
Enthält den Verschiebungsfaktor, um den die Figur in Y-Richtung verschoben werden soll
mz
Enthält den Verschiebungsfaktor, um den die Figur in Z-Richtung verschoben werden soll

Zeile 1
Prozeduranfang, Namensbezeichnung der Prozedur
Zeile 2 + 3
Ausgabe von Menuetext auf dem Bildschirm
Zeile 4
In Eingabe wird eine Liste von drei Werten eingelesen. Die Eingabe muß mit <ENTER> abgeschlossen werden
Zeile 5
Der Variable mx wird das erste Element der Eingabeliste zugewiesen
Zeile 6 + 7
Wie 5, jedoch für my (zweites Element) bzw. mz (drittes Element)
Zeile 8
Prozedur VERSCHIEBEN aufrufen
Zeile 9
Prozedurende

```
to menuelverschieben
pr [x,y,z Werte zum verschieben eingeben. Z.b 20 30 40
 < return >]
pr [Die Eingabe von 0 0 0 ergibt keine Verschiebung]
make "eingabe rl
make "mx item 1 :eingabe
make "my item 2 :eingabe
make "mz item 3 :eingabe
verschieben
end
```

Erläuterungen zum Programmteil MENUE1DREHEN:

MENUE1DREHEN ruft folgende Prozeduren auf:
DREHEN

MENUE1DREHEN wird von folgenden Prozeduren aufgerufen:
MENUE1

Verwendete Variablen sind:
eingabe
Liste von drei Werten, über Tastatur eingegeben
x
Drehwinkel um X-Achse
y
Drehwinkel um Y-Achse
z
Drehwinkel um Z-Achse

Zeile 1
Prozeduranfang, Namensbezeichnung der Prozedur
Zeile 2 + 3
Ausgabe von Menuetext auf den Bildschirm
Zeile 4
In **eingabe** wird eine Liste von drei Werten eingelesen. Die Eingabe muß mit <RETURN> abgeschlossen werden
Zeile 5
Der Variable x wird das erste Element der Liste zugewiesen
Zeile 6 + 7
wie 5, jedoch für y und z.
Zeile 8
Prozedur DREHEN aufrufen
Zeile 9
Prozedurende

```
to menueldrehen
pr [x,y,z Winkel zum drehen
 eingeben. Z.b 45 90 60
 < return >]
pr [Die Eingabe von 0 0 0
 ergibt keine Drehung]
make "eingabe rl
make "x item 1 :eingabe
make "y item 2 :eingabe
make "z item 3 :eingabe
drehen
end
```

Erläuterungen zum Programmteil KOORDINATENKREUZ:

KOORDINATENKREUZ ruft folgende Prozeduren auf:
MENUE1

KOORDINATENKREUZ wird von folgenden Prozeduren aufgerufen:
MENUE1

Verwendete Variablen sind: keine

Zeile 1
Prozeduranfang, Namensbezeichnung der Prozedur
Zeile 2
Die Schildkröte wird unsichtbar gemacht
Zeile 3
Der Zeichenstift der Schildkröte wird deaktiviert
Zeile 4
Die Schildkröte wird auf ihre Anfangskoordinaten (0 0) gesetzt
Zeile 5
Der Zeichenstift der Schildkröte wird aktiviert
Zeile 6 + 7
Die Y-Achse wird gezeichnet

Zeile 8
wie 4
Zeile 9 + 10
Die X-Achse wird gezeichnet
Zeile 11
wie 4
Zeile 12 + 13
Die Z-Achse wird gezeichnet
Zeile 14
wie 4
Zeile 15
Die Prozedur MENUE1 wird aufgerufen
Zeile 16
Prozedurende

```
to koordinatenkreuz
ht
pu
home
pd
setpos [0 270]
setpos [0 -270]
home
setpos [330 0]
setpos [-330 0]
home
setpos [270 270]
setpos [-270 -270]
home
menue1
end
```

Erläuterungen zum Programmteil MENUE2MODUS:

MENUE2MODUS ruft folgende Prozeduren auf: keine

MENUE2MODUS wird von folgenden Prozeduren aufgerufen:
MENUE2

Verwendete Variablen sind:
abfrage: Tastaturabfrage

Zeile 1
Prozeduranfang, Namensbezeichnung der Prozedur
Zeile 2 + 3
Ausgabe von Menuetext auf dem Bildschirm
Zeile 4
Ein Zeichen in **abfrage** einlesen. Die Tastatureingabe muß mit <RETURN> abgeschlossen werden
Zeile 5
Grafikbildschirm löschen?
Zeile 6
Figur stehen lassen? Schildkröte deaktivieren und (0 0) setzen
Zeile 7
Prozedurende

```
to menue2modus
pr [Bildschirm [L] oesche oder [S] tehen lassen]
pr []
make "abfrage rq
if or (:abfrage = "L) (:abfrage = "l) [cs]
if or (:abfrage = "S) (:abfrage = "s) [pu home]
end
```

Erläuterungen zum Programmteil MENUE2SPEICHERN:

MENUE2SPEICHERN ruft folgende Prozeduren auf: keine

MENUE2SPEICHERN wird von folgenden Prozeduren aufgerufen:
MENUE2

Verwendete Variablen sind: keine

Zeile 1
Prozeduranfang, Namensbezeichnung der Prozedur
Zeile 2 + 3
Ausgabe von Menuetext auf dem Bildschirm
Zeile 4
In dieser Zeile der Prozedur wird in rq ein Wort von der Tastatur eingelesen. Die Eingabe des Wortes muß mit <RETURN> abgeschlossen werden.
Dieses eingegebene Wort wird als Namenskennzeichnung der Bild-Datei genommen, die mit SAVEPIC abgespeichert wird
Zeile 5
Prozedurende

```
to menue2speichern
pr [Namen eingeben unter den das Bild abgespeichert
 werden soll.Laufwerk ist Standartlaufwerk]
pr [Z.b testbild oder grafik < return >]
savepic rq
menue2
end
```

Erläuterung zum Programmteil ZOOMEN:

ZOOMEN ruft folgende Prozeduren auf:
ZEICHNEN

ZOOMEN wird von folgenden Prozeduren aufgerufen:
MENUE1

Verwendete Variablen sind:
n
Wiederholungswert, wie oft die REPEAT-Schleife durchlaufen werden muß
k
Laufindex der Schleife
zw
Liste mit den X-Werten der zu berechnenden Punkte
yw
Liste mit den Y-Werten der zu berechnenden Punkte
zw
Liste mit den Z-Werten der zu berechnenden Punkte
xw1
Liste zur Zwischenspeicherung der neuen X-Werte
yw1
Liste zur Zwischenspeicherung der neuen Y-Werte
zw1
Liste zur Zwischenspeicherung der neuen Z-Werte
wert
Beinhaltet den veränderten Wert des aktuellen Punktes
zoomx
Enthält den Wert, um den in X-Richtung vergrößert wird
zoomy
Enthält den Wert, um den in Y-Richtung vergrößert wird
zoomz
Enthält den Wert, um den in Z-Richtung vergrößert wird

Zeile 1
Prozeduranfang, Namensgebung der Prozedur
Zeile 2
In n wird die Anzahl der Elemente, die in xw enthalten sind, gespeichert
Zeile 3
Der Laufindex k bekommt den Wert 1
Zeile 4 - 6
xw1, yw1 und zw1 werden zu leeren Listen gemacht

Zeile 7
Die Schleife wird n-mal wiederholt. In jedem Schleifendurchlauf bekommt die Variable wert das k-te Element der Liste xw zugeordnet, multipliziert mit dem Vergrößerungsfaktor zoomx. Die Hilfsvariable xw1 wird von hinten mit der Variable wert ergänzt. Der Laufindex k wird um 1 erhöht
Zeile 8
Die aktualisierten Werte in xw1 werden nach xw übertragen
Zeile 9
Der Laufindex k wird wieder auf 1 gesetzt
Zeile 10 - 12
Der Vorgang von Zeile 7-9 wiederholt sich hier entsprechend mit yw und yw1
Zeile 13 + 14
Wiederholung entspricht der mit zw und zw1
Zeile 15
Die Prozedur ZEICHNEN wird aufgerufen
Zeile 16
Prozedurende

```
to zoomen
make "n count :xw
make "k 1
make "xw1 []
make "yw1 []
make "zw1 []
repeat :n [make "wert (item :k :xw) * :zoomx make "xw1
 lput :wert :xw1 make "k :k + 1]
make "xw :xw1
make "k 1
repeat :n [make "wert (item :k :yw) * :zoomy make "yw1
 lput :wert :yw1 make "k :k + 1]
make "yw :yw1
make "k 1
repeat :n [make "wert (item :k :zw) * :zoomz make "zw1
 lput :wert :zw1 make "k :k + 1]
make "zw :zw1
zeichnen
end
```

Erläuterungen zum Programmteil VERSCHIEBEN:

VERSCHIEBEN ruft folgende Prozeduren auf:
ZEICHNEN

VERSCHIEBEN wird von folgenden Prozeduren aufgerufen:
MENUE1

Verwendete Variablen sind:
n
Anzahl Schleifendurchläufe
k
Laufindex innerhalb der Schleife
xw,yw, zw,xw1, yw1,zw1,
wert siehe ZOOMEN
mx,my, mz
Enthalten die Werte, um die in die entsprechende Richtungen verschoben wird

Zeile 1
Prozeduranfang, Namensbezeichnung der Prozedur
Zeile 2
In n wird die Anzahl der Elemente, die in xw enthalten sind, gespeichert
Zeile 3
Der Laufindex k bekommt den Wert 2
Zeile 4 - 6
In xw1, yw1 und zw1 wird jeweils eine Null eingetragen

Zeile 7
Die Schleife wird (n-1)-mal wiederholt. In jedem Schleifendurchlauf bekommt die Variable wert das k-te Element der Liste xw zugeordnet, addiert mit dem Verschiebungsfaktor mx. Die Hilfsvariable xw1 wird von hinten mit der Variable wert ergänzt. Der Laufindex k wird um 1 erhöht. Das 1-te Element der Liste in xw wird nicht bearbeitet, weshalb k=2 gesetzt wird. In xw1 wird das 1-te Element zu Null (Zeile 4)
Zeile 8
Die aktualisierten Werte in xw1 werden nach xw übertragen
Zeile 9
Der Laufindex k wird wieder auf 2 gesetzt
Zeile 10 - 12
Der Vorgang von Zeile 7 - 9 wiederholt sich hier entsprechend mit yw und yw1
Zeile 13 + 14
Der Vorgang von Zeile 7 - 9 wiederholt sich hier entsprechend mit zw und zw1
Zeile 15
Die Prozedur ZEICHNEN wird aufgerufen
16 Prozedurende

```
to verschieben
make "n count :xw
make "k 2
make "xw1 [0]
make "yw1 [0]
make "zw1 [0]
repeat :n - 1 [make "wert (item :k :xw) + :mx make "xw1
 lput :wert :xw1 make "k :k + 1]
make "xw :xw1
make "k 2
repeat :n - 1 [make "wert (item :k :yw) + :my make "yw1
 lput :wert :yw1 make "k :k + 1]
make "yw :yw1
make "k 2
repeat :n - 1 [make "wert (item :k :zw) + :mz make "zw1
 lput :wert :zw1 make "k :k + 1]
make "zw :zw1
zeichnen
end
```

Erläuterungen zum Programmteil DREHEN:

DREHEN ruft folgende Prozeduren auf:
ZEICHNEN
DREHENUNTER

DREHEN wird von folgenden Prozeduren aufgerufen:
MENUE1

Verwendete Variablen sind:
n
Anzahl Schleifendurchläufe
k
Laufindex der Schleife
xw,yw zw,xw1, yw1,zw1
wie gehabt

Zeile 1
Prozeduranfang, Namensbezeichnung der Prozedur
Zeile 2
In n wird die Anzahl der Elemente, die in xw enthalten sind, gespeichert
Zeile 3
Der Laufindex k bekommt den Wert 1
Zeile 4 - 6
xw1, yw1 und zw1 werden zu leeren Listen gemacht
Zeile 7
Die Schleife wird n-mal wiederholt. In jedem Durchlauf

TIPS & TRICKS

wird die Prozedur DREHENUNTER aufgerufen
Zeile 8 - 10
Die aktualisierten Werte von xw1, yw1, zw1 werden entsprechend nach xw, yw und zw übertragen
Zeile 11
Die Prozedur ZEICHNEN wird aufgerufen
Zeile 12
Prozedurende

```
to drehen
make "n count :xw
make "k 1
make "xw1 []
make "yw1 []
make "zw1 []
repeat :n [drehenunter]
make "xw :xw1
make "yw :yw1
make "zw :zw1
zeichnen
end
```

Erläuterungen zum Programmteil DREHENUNTER:

DREHENUNTER ruft folgende Prozeduren auf: keine

DREHENUNTER wird von folgenden Prozeduren aufgerufen:
DREHEN

Verwendete Variablen sind:
k
Laufindex
xw,yw, zw,xw1, yw1,zw1
wie gehabt
wert
Enthält den aktualisierten Wert der zu berechnenden Punkte
x
Enthält den Drehwinkel in X-Richtung
y
Enthält den Drehwinkel in Y-Richtung
z
Enthält den Drehwinkel in Z-Richtung
cox
Enthält den Cosinus von x
coy
Enthält den Cosinus von y
coz
Enthält den Cosinus von z
six
Enthält den Sinus von x
siy
Enthält den Sinus von y
siz
Enthält den Sinus von z

Zeile 1
Prozeduranfang, Namensbezeichnung der Prozedur
Zeile 2 - 7
Die Variablen cox, coy und coz erhalten die Cosinus-Werte von x, y und z. Entsprechen den Variablen six, siy und siz
Zeile 8 + 9
Hier wird der X-Wert des gedrehten Punktes berechnet
Zeile10
Der X-Wert wird in xw1 zwischengespeichert
Zeile 11 + 12
Entsprechend für den Y-Wert des gedrehten Punktes
Zeile 13 - 15
Hier wird der Z-Wert des gedrehten Punktes berechnet und zwischengespeichert
Zeile 16
Der Laufindex k wird um 1 erhöht
Zeile 17
Prozedurende

```
to drehenunter
make "cox cos :x
make "coy cos :y
make "coz cos :z
make "six sin :x
make "siy sin :y
make "siz sin :z
make "wert (item :k :xw) * (:cox * :coz - :six * :siy
 * :siz)
make "wert :wert + (item :k :yw) * (-1 * :six * :coz -
 :cox * :siy * :siz) + (item :k :zw  ) * -1 * :coy * :siz
make "xw1 lput :wert :xw1
make "wert (item :k :xw) * :six * :coy + (item :k :yw) *
 :cox * :coy + (item :k :zw) * -1   * :siy
make "yw1 lput :wert :yw1
make "wert (item :k :xw) * (:cox * :siz + :six * :siy *
 :coz)
make "wert :wert + (item :k :yw) * (-1 * :six * :siz +
 :cox * :siy * :coz) + (item :k :zw ) * :coy * :coz
make'"zw1 lput :wert :zw1
make "k :k + 1
end
```

Erläuterungen zum Programmteil ZEICHNEN:

ZEICHNEN ruft folgende Prozeduren auf:
ZEICHNENUNTER

ZEICHNEN wird von folgenden Prozeduren aufgerufen:
ZOOMEN
DREHEN
VERSCHIEBEN

Verwendete Variablen sind:
n
Anzahl Schleifendurchläufe
k
Laufindex
indvek
Enthält die Zuordnungsvorschrift, wie die eingegebenen Punkte verbunden werden müssen

Zeile 1
Prozeduranfang, Namensbezeichnung der Prozedur
Zeile 2
Die Schildkröte wird unsichtbar gemacht
Zeile 3
In n wird die Anzahl der Elemente, die in indvek enthalten sind, gespeichert
Zeile 4
Der Laufindex k bekommt den Wert 1
Zeile 5
Die Schleife wird (n-1)-mal wiederholt. In jedem Durchlauf wird die Prozedur ZEICHNENUNTER aufgerufen
Zeile 6
Prozedurende

```
to zeichnen
ht
make "n count :indvek
make "k 1
repeat :n - 1 [zeichnenunter]
end
```

Erläuterungen zum Programmteil ZEICHNENUNTER:

ZEICHNENUNTER ruft folgende Prozeduren auf: keine

ZEICHNENUNTER wird von folgenden Prozeduren aufgerufen:
ZEICHNEN

Verwendete Variablen sind:

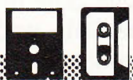
k
Laufindex
k1
Zeiger auf den Punkt, von dem aus gezeichnet wird
k2
Zeiger auf den Punkt, nach dem gezeichnet wird
ank
Ankathete des Dreiecks zwischen den Punkten, die durch k1 und k2 bestimmt werden
gegenk
Gegenkathete des Dreiecks zwischen den Punkten, die durch k1 und k2 bestimmt werden
hyp
Hypothenuse des Dreiecks zwischen den Punkten, die durch k1 und k2 bestimmt werden
wi
Winkel zwischen der Hypothenuse und der Horizontalen
kwi
Korrigierter Winkel, wie er von der LOGO-Prozedur SETH benötigt wird
xw,yw, zw
Listen mit X-, Y- und Z-Werten der Punkte
zm
Liste mit dem Zeichenmodus: auf welche Art die Punkte verbunden werden sollen

Zeile 1
Prozeduranfang, Namensbezeichnung der Prozedur
Zeile 2
k1 bekommt den k-ten Wert der Liste indvek zugewiesen, d.h. k1 zeigt auf den Punkt, von dem aus gezeichnet werden soll
Zeile 3
k2 bekommt den (k+1)-ten Wert der Liste indvek zugewiesen, d.h. k2 zeigt auf den Punkt, zu dem gezeichnet werden soll
Zeile 4
ank bekommt den Wert der Ankathete des Dreiecks zwischen den beiden Punkten, wobei jeweils den X-Werten das Sinus(45)-fache und den Y-Werten das Cosinus(45)-fache zugeordnet wird, was in beiden Fällen 0.707 ist
Zeile 5
Hier wird in gleicher Weise wie in Zeile 4 mit der Gegenkathete verfahren
Zeile 6
Hier wird die Ankathete auf 4 Stellen hinter dem Komma gerundet
Zeile 7
Hier wird die Gegenkathete auf 4 Stellen hinter dem Komma gerundet. Dieses muß durchgeführt werden, da die im folgenden zu berechnenden Winkel nicht zu sehr an den Wert 90 herankommen: z.B. 89.99999865437
Zeile 8
Wenn die Ankathete gleich Null ist, wird der Winkel der beiden Punkte zu 90 gemacht, denn die X-Koordinaten beider Punkte sind in diesem Fall gleich. In kwi wird der Winkel entsprechend der LOGO-Prozedur SETH korrigiert. Die Hypothenuse wird aus 'gegenk geteilt durch Sinus des Winkels' errechnet. Die Hypothenuse ist gleichbedeutend der Entfernung der beiden Punkte
Zeile 9
Wenn die Gegenkathete gleich Null ist, so wird der Winkel der beiden Punkte zu Null gemacht, denn die Y-Koordinaten beider Punkte sind in diesem Fall gleich. kwi wird wieder entsprechend korrigiert. Die Hypothenuse errechnet sich hier aus 'ank geteilt durch Cosinus des Winkels'. Die in Zeile 8 und 9 beschriebenen Fälle sind Sonderfälle und müssen vor der weiteren Prozedurausführung abgefangen werden
Zeile 10
Wenn ank größer Null und gegenk ungleich Null ist, so wird der Winkel aus 'Arcustangens (gegenk) geteilt durch ank' errechnet. In kwi wird der Winkel korrigiert. Die Hypothenuse errechnet sich aus 'ank geteilt durch Cosinus des Winkels'. Wenn die Hypothenuse negativ ist, so wird sie durch Multiplikation mit -1 positiv gemacht
Zeile 11
Wenn ank kleiner Null und gegenk ungleich Null ist, so wird entsprechend wie in Zeile 10 verfahren, kwi wird jedoch anders korrigiert
Zeile 12
Die Schildkröte wird entsprechend des korrigierten Winkels kwl auf die Gradposition gesetzt
Zeile 13
Der Zeichenstift der Schildkröte wird wieder aktiviert
Zeile 14
Wenn das k-te Element der Liste zm eine Null (nicht zeichnen) ist, so wird der Zeichenstift der Schildkröte wieder deaktiviert. Ist das Element eine Eins, so bleibt der Zeichenstift aktiviert.
Zeile 15
Es wird eine Linie vom Punkt, der durch k1 bezeichnet wird, nach dem Punkt, der durch k2 bezeichnet wird, gezogen
Zeile 16
Der Laufindex k wird um 1 erhöht
Zeile 17
Prozedurende

```
to zeichnenunter
make "k1 item :k :indvek
make "k2 item :k + 1 :indvek
make "ank (((item :k2 :xw) + 0.707 * (item :k2 :zw)) -
 ((item :k1 :xw) + 0.707 * (item :k1 :zw)))
make "gegenk (((item :k2 :yw) + 0.707 * (item :k2 :zw)) -
 ((item :k1 :yw) + 0.707 * (item:k1 :zw)))
make "ank (round (:ank * 10000) / 10000
make "gegenk (round (:gegenk * 10000) / 10000
if :ank = 0 [make "wi 90 make "kwi 90 - :wi make "hyp
 :gegenk / (sin :wi)]
if :gegenk = 0 [make "wi 0 make "kwi 90 - :wi make "hyp
 :ank / (cos :wi)]
if and (:ank > 0) (not (:gegenk = 0)) [make "wi arctan
 (:gegenk / :ank) make "kwi 90 - :w
i make "hyp :ank / (cos :wi) if :hyp < 0 [make "hyp :hyp
 * (-1)] []] []
if and (:ank < 0) (not (:gegenk = 0)) [make "wi arctan
 (:gegenk / :ank) make "kwi 270 - :
wi make "hyp :ank / (cos :wi) if :hyp < 0 [make "hyp :hyp
 * (-1)] []] []
seth :kwi
pd
if (item :k :zm) = 0 [pu] []
fd :hyp
make "k :k + 1
end
```

(Thomas Schulte)

The Designer



The Designer ist ein hervorragendes Grafikprogramm mit vielen Optionen, die dieses Tool in professionelle Gefilde heben.

Das Programm ist vollkommen menuegesteuert und benötigt keine Tastatureingaben, außer bei Texten. Anstelle eines Joysticks können die Cursortasten + Copy verwandt werden.

Abfragen, die mit JA oder NEIN beantwortet werden müssen, können durch Joystick links (Nein) oder Joystick rechts (Ja) angewählt werden.

Bei Zahlwerten (z.B. Zeichengeschwindigkeit) gilt das gleiche:
Joystick links = kleinere Zahl
Joystick rechts = größere Zahl

Die Bestätigung einer Eingabe erfolgt über die Feuer- bzw. Copy-Taste.

Während des Zeichnens ist darauf zu achten, daß die Cursorgeschwindigkeit bei anhaltendem Tastendruck zunimmt. So kann man schnelle und präzise Bewegungen vornehmen.

Am unteren Bildrand befindet sich ein Farbbalken sowie der Schriftzug MENUE. Durch Anfahren mit dem Cursor und Betätigen der Auslösetaste kann man so den Farbstift wechseln oder ins Hauptmenue zurückkehren.

Im Hauptmenue hat man folgende Optionen zur Verfügung:

Farben: Hiermit können die INK's selektiert werden.
Modus: Der Bildschirmmodus (1, 2, 3) wird festgelegt.
Pinsel: Die Pinselbreite wird festgelegt. Dies hat nur Einfluß auf die Menuepunkte Malen, Linie, Linien und Strahlen.
Geschw. (Geschwindigkeit): Die Zeichengeschwindigkeit wird grob festgelegt.
XOR: Das Gezeichnete wird mit dem schon vorhandenen Bildschirminhalt verknüpft (Exclusiv-Oder-Verknüpfung). Dies gilt nicht bei den Funktionen Malen, Kreis, Ellipse, Füllen und Kopieren. Bei ausgefüllten Dreiecken können durch diese Funktion interessante Muster entstehen.

Paramet. (Parameter): Verschiedene Parameter für die Abspeicherung eines Bildes können eingestellt werden. Wenn man die Frage "Bildparameter abspeichern?" mit Ja beantwortet, werden der aktuelle Mode und die INK's vor dem Bild als Basic-Programm abgespeichert. Diese Files können direkt mit RUN"Name" gestartet werden. Außerdem werden beim Laden eines Bildes Parameter mit eingelesen. Hat man nur die Bild-Daten abgespeichert, muß vor dem Laden die Frage "Bildparameter abspeichern?" mit Nein beantwortet werden, da kein Parameterfile existiert.
Bei Kassettenbetrieb kann auch die Baudrate und das Aufzeichnungsformat (Normal/Fast) gewählt werden. Bei "Fast" wird das gesamte Bild in einem Block abgespeichert.

Malen: Durch Drücken der Feuertaste kann nun ein Punkt gesetzt werden.
Linie: Das Programm zeichnet zwischen zwei, durch Drücken des Feuerknopfes, gesetzten Punkten eine Linie.
Linien: Der Endpunkt der zuletzt gezeichneten Linie ist zugleich der Anfangspunkt der nächsten Linie.
Strahlen: Der erste gesetzte Punkt ist zugleich der Anfangspunkt aller weiteren Linien.
Kreis: Zuerst wird der Mittelpunkt des Kreises durch Drücken der Feuertaste festgelegt, danach kann der Radius gewählt und durch ein erneutes Drücken der Feuertaste bestätigt werden.
Ellipse: Der Mittelpunkt wird wieder durch Drücken der Feuertaste festgelegt, danach muß sowohl der x- als auch der y-Radius eingestellt und durch Drücken der Feuertaste bestätigt werden.
Rechteck: Die zwei diagonal gegenüberliegenden Eckpunkte müssen gewählt und jeweils durch Drücken der Feuertaste bestätigt werden.
Dreieck: Hier müssen alle drei Eckpunkte gewählt und jeweils durch Drücken des Feuerknopfes bestätigt werden.
Füllen: Hiermit können geschlossene Flächen mit der vorher gewählten Farbe ausgefüllt werden, indem der Cursor in die Fläche hineingesteuert wird und nun durch Drücken der Feuertaste der Vorgang gestartet wird.

Kopieren: Nach Auswahl dieser Funktion fragt das Programm, ob der Hintergrund, auf den der Bildschirmausschnitt kopiert werden soll, gelöscht werden soll. Der rechteckige Bildschirmausschnitt, der wie in der Funkion Rechteck gewählt wird, kann nun an jede beliebige Stelle geschoben werden. Das Drücken der Feuertaste startet den Kopiervorgang.
Zoom: Ein kleiner Ausschnitt des Bildschirms, der vergrößert dargestellt wird, kann mit Hilfe der Feuertaste Punkt für Punkt verändert werden. Diese Änderungen werden in das Bild übernommen, indem man mit dem Cursor das Menuefeld ansteuert und dort den Feuerknopf drückt.
Text: Hier kann nun ein Text eingegeben werden und an jeder Stelle des Bildschirmes plaziert werden.
Spray: Wenn Sie in dieser Funktion den Feuerknopf drücken, so werden einzelne Punkte unter dem Grafik-Cursor gezeichnet.
Löschen: Mit dieser Funktion wird das gesamte Bild gelöscht.
Save: Das Bild, das sich momentan im Speicher befindet, wird abgespeichert.
Load: Diese Funktion lädt ein Bild in den Speicher.
Ende: Diese Funktion ermöglicht eine Rückkehr in Basic.
Neustart: Alle Parameter werden auf die Anfangswerte gestellt.

(O. Heggelbacher/TM)

Listing 1

```
100 '******************************         [1285]
110 '*                            *         [175]
120 '*        The Designer        *         [904]
130 '*                            *         [175]
140 '*   (C) 1986 Andromeda-Soft  *         [1428]
150 '*                            *         [175]
160 '*      geschrieben von       *         [2112]
170 '*                            *         [175]
180 '*    Oliver Heggelbacher     *         [1056]
190 '*                            *         [175]
200 '****** CPC 464/664/6128 ******         [605]
210 '                                       [117]
220 SYMBOL AFTER 240:MEMORY &5FFF:OPENOUT"  [7062]
dummy":MEMORY HIMEM-1:CLOSEOUT:SYMBOL 240,
20,81,68,146,73,138,37,72
```

```
230 ad=&A040                                    [746]
240 FOR t=0 TO 29:su=0                          [1008]
250     FOR i=0 TO 15                           [504]
260         READ w$:w=VAL("&"+w$)               [1166]
270         POKE ad,w:ad=ad+1                   [546]
280         su=su+w                             [92]
290     NEXT i                                  [375]
300     READ c$:c=VAL("&"+c$)                   [1443]
310     IF c<>su THEN PRINT "Data-Fehler in     [3935]
 Zeile";370+t*10:END
320 NEXT t                                      [360]
330 CALL &A040:MODE 1:|PUT                      [977]
340 INPUT "CPC 664/6128 (J/N)";a$:a$=UPPER      [5630]
$(a$):IF a$="J" THEN POKE &A300,1 ELSE IF
a$="N" THEN POKE &A300,0 ELSE 340
350 RUN"!designer.prg"                          [974]
360 '                                           [117]

370 DATA 21,49,A0,01,4D,A0,C3,D1,BC,00,00,      [2767]
00,00,64,A0,C3 , 60F
380 DATA 81,A0,C3,8A,A0,C3,93,A0,C3,9C,A0,      [4057]
C3,A5,A0,C3,B1 , A7F
390 DATA A0,C3,D3,A0,48,53,41,56,C5,48,4C,      [2386]
4F,41,C4,53,41 , 749
400 DATA 56,C5,4C,4F,41,C4,47,45,D4,50,55,      [2426]
D4,46,49,4C,CC , 73B
410 DATA 00,21,00,A0,11,40,00,C3,9E,BC,21,      [1670]
00,A0,11,40,00 , 441
420 DATA C3,A1,BC,21,00,60,11,00,40,C3,9E,      [2933]
BC,21,00,60,11 , 5A1
430 DATA 00,40,C3,A1,BC,21,00,60,11,00,C0,      [2733]
01,00,40,ED,B0 , 590
440 DATA C9,21,00,C0,11,00,60,01,00,40,ED,      [1959]
B0,C9,00,00,00 , 4C2
450 DATA 00,00,00,00,00,00,00,00,00,00,00,      [1328]
00,00,11,00,00 , 011
460 DATA C3,1E,A1,FE,04,C0,DD,6E,00,DD,66,      [2332]
01,22,C7,A0,DD , 839
470 DATA 7E,02,32,CC,A0,DD,6E,04,DD,66,05,      [3596]
DD,5E,06,DD,56 , 729
480 DATA 07,ED,53,BF,A0,22,C1,A0,21,00,50,      [1984]
22,BD,A0,ED,5B , 761
490 DATA BF,A0,2A,C1,A0,D5,E5,CD,1D,BC,7E,      [3703]
A1,32,CB,A0,E1 , 9E7
500 DATA D1,7A,B3,28,B8,1B,3A,CB,A0,B7,CA,      [2008]
05,A1,13,ED,53 , 818
510 DATA C3,A0,13,3E,01,32,C9,A0,32,CA,A0,      [2412]
ED,53,BF,A0,22 , 7AD
520 DATA C1,A0,CD,1D,BC,7E,A1,B7,20,63,ED,      [2288]
5B,BF,A0,2A,C1 , 8F2
530 DATA A0,23,CD,1D,BC,7E,A1,21,C9,A0,46,      [4041]
77,B7,20,10,78 , 72E
540 DATA B7,28,0C,ED,5B,BF,A0,ED,4B,C1,A0,      [3504]
03,CD,FE,A1,ED , 987
550 DATA 5B,BF,A0,2A,C1,A0,2B,CD,1D,BC,7E,      [3630]
A1,21,CA,A0,46 , 806
560 DATA 77,B7,20,10,78,B7,28,0C,ED,5B,BF,      [2794]
A0,ED,4B,C1,A0 , 801
570 DATA 0B,CD,FE,A1,ED,5B,BF,A0,2A,C1,A0,      [3235]
13,22,C5,A0,2A , 86D
580 DATA C7,A0,B7,ED,52,38,06,2A,C5,A0,C3,      [3461]
2B,A1,2A,C3,A0 , 846
590 DATA ED,5B,C1,A0,7C,B5,28,01,23,CD,E8,      [2632]
A1,CD,C0,BB,2A , 8EE
600 DATA BF,A0,ED,5B,C1,A0,2B,CD,E8,A1,CD,      [2821]
F6,BB,2A,BD,A0 , A8E
610 DATA 01,04,00,B7,ED,42,22,BD,A0,7C,FE,      [3126]
4F,C8,5E,23,56 , 6D2
620 DATA 23,4E,23,46,69,60,ED,53,BF,A0,22,      [2765]
C1,A0,CD,1D,BC , 76B
630 DATA 7E,A1,B7,20,D8,C3,FE,A0,3A,CC,A0,      [3100]
47,05,28,0C,22 , 777
640 DATA C5,A0,C5,ED,4B,C5,A0,09,C1,10,F7,      [2395]
EB,29,C9,3E,07 , 8BA
650 DATA B9,C8,3E,C8,B9,C8,2A,BD,A0,73,23,      [3263]
72,23,71,23,70 , 7BE
660 DATA 23,22,BD,A0,C9,00,00,00,00,00,00,      [2692]
00,00,00,00,00 , 26B

670 MEMORY &A03F:ad=&A040                       [925]
680 FOR t=0 TO 29:su=0                          [1008]
690     FOR i=0 TO 15                           [504]
700         READ w$:w=VAL("&"+w$)               [1166]
710         POKE ad,w:ad=ad+1                   [546]
720         su=su+w                             [92]
730     NEXT i                                  [375]
740     READ c$:c=VAL("&"+c$)                   [1443]
750     IF c<>su THEN PRINT "Data-Fehler in     [3726]
 Zeile";120+t*10:END
760 NEXT t                                      [360]
```

Listing 2

```
10 GOTO 330                                     [506]
20 "                                            [102]
30 " The Designer - Hauptprogramm               [2093]
40 "                                            [102]
50 "      $ 1986 Andromeda-Soft                 [2674]
60 "                                            [102]
70 '                                            [117]
80 ' Unterprogramme                             [916]
90 rx=0:ry=0:fi=0:sp=0                          [1154]
100 a=JOY(0):IF a AND 1 OR NOT INKEY(0)THE      [3608]
N ry=1
110 IF a AND 2 OR NOT INKEY(2)THEN ry=-1        [2287]
120 IF a AND 4 OR NOT INKEY(8)THEN rx=-1        [1679]
130 IF a AND 8 OR NOT INKEY(1)THEN rx=1         [586]
140 IF a AND 16 OR NOT INKEY(9)THEN fi=1        [940]
150 IF NOT INKEY(47)THEN sp=1                   [1307]
160 IF rx=0 AND ry=0 AND fi+sp=0 THEN gx=0      [2877]
:gy=0:GOTO 100
170 RETURN                                      [555]
180 '                                           [117]
190 MOVE x,y-8:DRAW x,y,fa:MOVER ab,-2:DRA      [12901]
W x+16,y-12:MOVE x+ab,y:DRAW x+8,y:x2=x:y2
=y:GOSUB 90:gx=gx+rx:g =gy+ry:IF rx=0 THEN
 gx=0 ELSE IF SGN(gx)<>SGN(rx)THEN gx=0
200 IF ry=0 THEN gy=0 ELSE IF SGN(gy)<>SGN      [3093]
(ry)THEN gy=0
210 x=x+gx/ug:y=y+gy/ug:IF x<0 THEN x=0 EL      [4175]
SE IF x>639 THEN x=639
220 IF y<2 THEN y=2 ELSE IF y>399 THEN y=3      [2266]
99
230 MOVE x2,y2-8:DRAW x2,y2,fa:MOVER ab,-2      [5519]
:DRAW x2+16,y2-12:MOVE x2+ab,y2:DRAW x2+8,
y2:IF f2+fi+sp+fl=0 THEN 190
240 IF sp=1 THEN RETURN                         [1842]
250 IF fi=1 THEN 270                            [486]
260 IF fl=1 THEN RETURN                         [674]
270 IF y>15 THEN RETURN                         [441]
280 IF x<479 THEN zf=TEST(x,y-2):BORDER co      [5886]
(zf):x=x2:y=y2:GOTO 190 ELSE fi=0:f2=1:RET
URN
290 :                                           [174]
300 :                                           [174]
310 '                                           [117]
320 ' Initialisierung                           [1300]
330 DEFINT a-s,w-z                              [642]
340 DIM co(15):FOR t=0 TO 15:READ co(t):NE      [2835]
XT t
350 DATA 0,26,24,23,22,20,18,17,16,15,13,1      [1818]
2,9,8,6,2
360 mo=0:ps=0:ab=4:af=15:zf=2:ge=8:ux=1:uy      [3743]
=1:fa=1:dl$="N"
370 k$="N":e$="N":r$="N":d$="N":c$="N":xo=      [2859]
0:ba=1:fo$="Standard":pa$="N":pic$="*.pic"
380 DIM zo(31,23)                               [1153]
390 ENV 1,1,13,1,13,-1,5                        [974]
400 '                                           [117]
410 PRINT CHR$(23);"0";:BORDER 0:INK 0,0:I      [5190]
NK 1,17:INK 2,24:INK 3,11,22:SPEED INK 20,
20
420 MODE 1:PEN 1:LOCATE 9,1:PRINT"T H E         [4120]
D E S I G N E R"
430 PEN 2:LOCATE 9,3:PRINT"(C) 1986 Androm      [2979]
eda-Soft"
440 PEN 1:PRINT:PRINT STRING$(40,154)           [1935]
450 ZONE 10:PEN 2                               [976]
460 LOCATE 2,10:PRINT"FARBEN"," MODUS"," P      [4735]
INSEL"," GESCHW."
470 LOCATE 2,12:PRINT"XOR"," PARAMET."," M      [4838]
ALEN"," LINIE"
480 LOCATE 2,14:PRINT"LINIEN"," STRAHLEN",      [3809]
" KREIS"," ELLIPSE"
490 LOCATE 2,16:PRINT"RECHTECK"," DREIECK"      [3504]
," FUELLEN"," KOPIEREN"
500 LOCATE 2,18:PRINT"ZOOM"," TEXT"," SPRA      [2818]
Y"," LOESCHEN"
510 LOCATE 2,20:PRINT"SAVE"," LOAD"," ENDE      [3626]
"," NEUSTART"
520 mx=ux:my=uy                                 [996]
530 PLOT mx*160-148,290-my*32,3:DRAWR 140,      [7429]
0:DRAWR 0,-20:DRAWR-140,0:DRAWR 0,20:mx2=m
x:my2=my:GOSUB 90
540 ux=ux+rx/5:IF ux<1 OR ux>4 THEN ux=mx2      [2251]
550 uy=uy-ry/5:IF uy<1 OR uy>6 THEN uy=my2      [1838]
560 mx=ux:my=uy:IF fi=1 OR mx<>mx2 OR my<>      [6638]
my2 THEN PLOT mx2*160-148,290-my2*32,0:DRA
WR 140,0:DRAWR 0,-20:DRAWR-140,0:DRAWR 0,2
0
```

```
570 IF fi=0 THEN 530 ELSE SOUND 130,119,0, [1511]
0,1
580 WHILE JOY(0)>15:WEND [1549]
590 WHILE INKEY$<>"":WEND [1786]
600 zw=mx+(my-1)*4:ON zw GOSUB 640,720,810 [7747]
,870,930,990,1180,1230,1330,1430,1530,1740
,1990,2170,2450,2540,2780,2990,3080,3130,3
200,3340,3540
610 IF zw=24 THEN RUN ELSE 410 [1259]
620 ' [117]
630 ' Farben [742]
640 MODE 0:FOR t=0 TO af:INK t,co(t):PEN t [5100]
:PRINT STRING$(10,143):NEXT t
650 y=1 [354]
660 f=y-1:LOCATE 12,y:PEN fa:PRINT CHR$(24 [2950]
2);:GOSUB 90:IF fi=1 THEN RETURN
670 co(f)=co(f)+rx:LOCATE 12,y:PRINT" ":y= [5842]
y-ry:IF co(f)<0 THEN co(f)=0 ELSE IF co(f)
>26 THEN co(f)=26
680 IF y<1 THEN y=1 ELSE IF y>af+1 THEN y= [1960]
af+1
690 INK f,co(f):FOR t=1 TO 50:NEXT t:GOTO [2656]
660
700 ' [117]
710 ' Mode [775]
720 CLS:PRINT"Mode (0-2) :";mo [1409]
730 GOSUB 3720:GOSUB 90:IF fi=1 THEN 760 [2998]
740 mo=mo+rx:IF mo<0 THEN mo=0 ELSE IF mo> [3067]
2 THEN mo=2
750 LOCATE 13,1:CALL &BD19:PRINT mo:GOTO 7 [2010]
30
760 ab=2^(2-mo):af=(2^ab)-1:IF zf>af THEN [2571]
zf=af
770 IF fa>af THEN fa=af [868]
780 RETURN [555]
790 ' [117]
800 ' Pinsel [85]
810 CLS [91]
820 LOCATE 1,1:PRINT"Pinselstaerke <"ps"> [3461]
":GOSUB 90:IF fi=1 THEN RETURN
830 ps=ps+rx*ab:IF ps<0 THEN ps=0 ELSE IF [2432]
ps>64 THEN ps=64
840 GOTO 820 [320]
850 ' [117]
860 ' Zeichengeschw. [1789]
870 CLS [91]
880 LOCATE 1,1:PRINT"Zeichengeschwindigkei [5259]
t <"ge"> ":GOSUB 90:IF fi=1 THEN RETURN
890 ge=ge+rx:IF ge<1 THEN ge=1 ELSE IF ge> [2103]
15 THEN ge=15
900 GOTO 880 [548]
910 ' [117]
920 ' XOR-Funkt. [347]
930 CLS:IF xo=0 THEN x$="N"ELSE x$="J" [2358]
940 LOCATE 1,1:PRINT"XOR-Funktion <"x$">": [3498]
GOSUB 90:IF fi=1 THEN RETURN
950 IF rx=-1 THEN x$="N":xo=0 ELSE IF rx=1 [4505]
THEN x$="J":xo=1
960 GOTO 940 [312]
970 ' [117]
980 ' Parameter [726]
990 CLS [91]
1000 LOCATE 1,1:PRINT"Bildparameter abspei [6169]
chern <"pa$">":GOSUB 90:IF fi=1 THEN 1030
1010 IF rx=-1 THEN pa$="N"ELSE IF rx=1 THE [1793]
N pa$="J"
1020 GOTO 1000 [339]
1030 GOSUB 3720 [1025]
1040 LOCATE 1,3:PRINT"Diskettenlaufwerk <" [5528]
dl$">":GOSUB 90:IF fi=1 THEN 1070
1050 IF rx=-1 THEN dl$="N"ELSE dl$="J" [1157]
1060 GOTO 1040 [363]
1070 IF dl$="J"THEN RETURN [1802]
1080 GOSUB 3720 [1025]
1090 LOCATE 1,5:PRINT"BAUD-Rate :";1000+ba [5552]
*1000:GOSUB 90:IF fi=1 THEN 1120
1100 ba=ba+rx:GOSUB 3720:IF ba<0 OR ba>3 T [2279]
HEN ba=ba-rx
1110 GOTO 1090 [393]
1120 GOSUB 3720 [1025]
1130 LOCATE 1,7:PRINT"Format :";fo$;SPACE$ [4729]
(4):GOSUB 90:IF fi=1 THEN RETURN
1140 IF rx=-1 THEN fo$="Standard"ELSE IF r [2697]
x=1 THEN fo$="Fast"
1150 GOTO 1130 [345]
1160 ' [117]
1170 ' Malen [1072]
1180 GOSUB 3580 [1005]
1190 GOSUB 190:GOSUB 3740:IF f2+sp=0 THEN [8363]
PRINT CHR$(23);"0";:FOR t=0 TO ps STEP ab:
MOVE x2+t,y2:DRAW x+t,y,zf:NEXT t:PRINT CH
R$(23);"1";:GOTO 1190
1200 IF sp=1 THEN 1190 ELSE 3700 [1535]
1210 ' [117]
1220 ' Linie [1049]
1230 GOSUB 3580 [1005]
1240 GOSUB 190:IF f2=1 THEN 3700 [1773]
1250 GOSUB 3720:GOSUB 3740:IF sp=1 THEN 12 [3007]
40 ELSE fl=1
1260 ax=x:ay=y [872]
1270 MOVE ax,ay:DRAW x,y,fa:GOSUB 190:MOVE [6732]
 ax,ay:DRAW x2,y2,fa:GOSUB 3740:IF fi+f2=0
 THEN 1270
1280 GOSUB 3720 [1025]
1290 IF f2=1 THEN 3700 [1469]
1300 PRINT CHR$(23);CHR$(xo);:FOR t=0 TO p [6951]
s STEP ab:MOVE ax+t,ay:DRAW x+t,y,zf:NEXT
t:PRINT CHR$(23);"1";:fl=0:GOTO 1240
1310 ' [117]
1320 ' Linien [1276]
1330 GOSUB 3580 [1005]
1340 GOSUB 190:IF f2=1 THEN 3700 [1773]
1350 GOSUB 3720:GOSUB 3740:IF sp=1 THEN 13 [2755]
40 ELSE fl=1
1360 ax=x:ay=y [872]
1370 MOVE ax,ay:DRAW x,y,fa:GOSUB 190:MOVE [6928]
 ax,ay:DRAW x2,y2,fa:GOSUB 3740:IF fi+f2=0
 THEN 1370
1380 GOSUB 3720 [1025]
1390 IF f2=1 THEN 3700 [1469]
1400 PRINT CHR$(23);CHR$(xo);:FOR t=0 TO p [8166]
s STEP ab:MOVE ax+t,ay:DRAW x+t,y,zf:NEXT
t:PRINT CHR$(23);"1";:ax=x:ay=y:GOTO 1370
1410 ' [117]
1420 ' Strahlen [128]
1430 GOSUB 3580 [1005]
1440 GOSUB 190:IF f2=1 THEN 3700 [1773]
1450 GOSUB 3720:GOSUB 3740:IF sp=1 THEN 14 [3356]
40 ELSE fl=1
1460 ax=x:ay=y [872]
1470 MOVE ax,ay:DRAW x,y,fa:GOSUB 190:MOVE [6996]
 ax,ay:DRAW x2,y2,fa:GOSUB 3740:IF fi+f2=0
 THEN 1470
1480 GOSUB 3720 [1025]
1490 IF f2=1 THEN 3700 [1469]
1500 PRINT CHR$(23);CHR$(xo);:FOR t=0 TO p [7461]
s STEP ab:MOVE ax+t,ay:DRAW x+t,y,zf:NEXT
t:PRINT CHR$(23);"1";:GOTO 1470
1510 ' [117]
1520 ' Kreis [373]
1530 CLS [91]
1540 LOCATE 1,1:PRINT"Ausfuellen <"k$">":G [5207]
OSUB 90:IF fi=1 THEN 1570
1550 IF rx=-1 THEN k$="N"ELSE IF rx=1 THEN [1622]
 k$="J"
1560 GOTO 1540 [391]
1570 GOSUB 3580 [1005]
1580 GOSUB 190:IF f2=1 THEN 3700 [1773]
1590 ax=x:ay=y [872]
1600 GOSUB 3720:GOSUB 3740:IF sp=1 THEN 15 [3793]
80 ELSE fl=1
1610 MOVE ax,ay:DRAW x,ay,fa:GOSUB 190:MOV [3379]
E ax,ay:DRAW x2,ay,fa:GOSUB 3740:IF fi+f2=
0 THEN 1610
1620 IF f2=1 THEN 3700 [1469]
1630 GOSUB 3720 [1025]
1640 r=ABS(ax-x)/1.14 [577]
1650 ORIGIN ax,ay,0,639,16,399:vr=r*r [2810]
1660 PRINT CHR$(23);"0";:PLOT-330,-220,zf [2433]
1670 IF k$="J"THEN 1700 [1268]
1680 FOR v=0 TO r/1.414 STEP ab/4:a=SQR(vr [9603]
-v*v):PLOT-a*1.14,-v:PLOT-a*1.14,v:PLOT a*
1.14,-v:PLOT a*1.14,v:PLOT-v*1.14,-a:PLOT-
v*1.14,a:PLOT v*1.14,-a:PLOT v*1.14,a:NEXT
v
1690 ORIGIN 0,0,0,639,0,399:PRINT CHR$(23) [1714]
;"1";:fl=0:GOTO 1580
1700 FOR v=0 TO r/1.414 STEP ab/4:a=SQR(vr [9982]
-v*v):MOVE-a*1.14,-v:DRAW a*1.14,-v:MOVE-a
*1.14,v:DRAW a*1.14,v:MOVE-v*1.14,-a:DRAW
v*1.14,-a:MOVE-v*1.14,a:DRAW v*1.14,a:NEXT
v
1710 ORIGIN 0,0,0,639,0,399:PRINT CHR$(23) [1714]
;"1";:fl=0:GOTO 1580
1720 ' [117]
1730 ' Ellipse [490]
1740 CLS [91]
1750 LOCATE 1,1:PRINT"Ausfuellen <"e$">":G [3393]
```

ANWENDUNG

```
OSUB 90:IF fi=1 THEN 1780
1760 IF rx=-1 THEN e$="N"ELSE IF rx=1 THEN [1022]
 e$="J"
1770 GOTO 1750                                    [349]
1780 GOSUB 3580                                   [1005]
1790 GOSUB 190:IF f2=1 THEN 3700                  [1773]
1800 ax=x:ay=y                                    [872]
1810 GOSUB 3720:GOSUB 3740:IF sp=1 THEN 17 [3243]
90 ELSE fl=1
1820 MOVE ax,ay:DRAW x,ay,fa:GOSUB 190:MOV [3752]
E ax,ay:DRAW x2,ay,fa:GOSUB 3740:IF fi+f2=
0 THEN 1820
1830 IF f2=1 THEN 3700                            [1469]
1840 GOSUB 3720                                   [1025]
1850 xr=ABS(ax-x)                                 [1501]
1860 MOVE ax,ay:DRAW ax,y,fa:GOSUB 190:MOV [4923]
E ax,ay:DRAW ax,y2,fa:GOSUB 3740:IF fi+f2=
0 THEN 1860
1870 IF f2=1 THEN 3700                            [1469]
1880 yr=ABS(ay-y):IF yr=0 THEN yr=1               [1172]
1890 vr=xr/yr:r=yr:ur=r*r                         [1702]
1900 PLOT-10,-10,zf:ORIGIN ax,ay,0,639,16, [2425]
399:PRINT CHR$(23);"0";
1910 IF xr>yr THEN us=yr/xr ELSE us=0.5           [3300]
1920 IF e$="J"THEN 1950                           [892]
1930 FOR v=0 TO SQR(ur)/1.414 STEP us:u=SQ [9755]
R(ur-v*v):PLOT-u*vr,-v:PLOT-u*vr,v:PLOT u*
vr,-v:PLOT u*vr,v:PLOT-v*vr,-u:PLOT-v*vr,u
:PLOT v*vr,-u:PLOT v*vr,u:NEXT v
1940 ORIGIN 0,0,0,639,0,399:PRINT CHR$(23) [1712]
;"1";:fl=0:GOTO 1790
1950 FOR v=0 TO SQR(ur)/1.414 STEP us:u=SQ [11836]
R(ur-v*v):MOVE-u*vr,-v:DRAW u*vr,-v:MOVE-u
*vr,v:DRAW u*vr,v:MOVE-v*vr,-u:DRAW v*vr,-
u:MOVE-v*vr,u:DRAW v*vr,u:NEXT v
1960 ORIGIN 0,0,0,639,0,399:PRINT CHR$(23) [1712]
;"1";:fl=0:GOTO 1790
1970 '                                            [117]
1980 ' Rechteck                                   [786]
1990 CLS                                          [91]
2000 LOCATE 1,1:PRINT"Ausfuellen <"r$">":G [4475]
OSUB 90:IF fi=1 THEN 2030
2010 IF rx=-1 THEN r$="N"ELSE IF rx=1 THEN [2676]
 r$="J"
2020 GOTO 2000                                    [363]
2030 GOSUB 3580                                   [1005]
2040 GOSUB 190:IF f2=1 THEN 3700                  [1773]
2050 GOSUB 3720:GOSUB 3740:IF sp=1 THEN 20 [3340]
40 ELSE fl=1
2060 ax=x:ay=y                                    [872]
2070 MOVE ax,ay:DRAW ax,y,fa:DRAW x,y:DRAW [9049]
 x,ay:DRAW ax,ay:GOSUB 190:MOVE ax,ay:DRAW
 ax,y2,fa:DRAW x2,y2:DRAW x2,ay:DRAW ax,ay
:GOSUB 3740:IF fi+f2=0 THEN 2070
2080 IF f2=1 THEN 3700                            [1469]
2090 GOSUB 3720                                   [1025]
2100 PLOT-10,-10,zf:PRINT CHR$(23);CHR$(xo [1560]
);
2110 IF r$="J"THEN 2130                           [606]
2120 MOVE ax,ay:DRAW ax,y:DRAW x,y:DRAW x, [3334]
ay:DRAW ax,ay:PRINT CHR$(23);"1";:fl=0:GOT
O 2040
2130 IF x<ax THEN a=x:b=ax ELSE a=ax:b=x          [2647]
2140 FOR t=a TO b STEP ab:MOVE t,ay:DRAW t [4533]
,y:NEXT t:PRINT CHR$(23);"1";:fl=0:GOTO 20
40
2150 '                                            [117]
2160 ' Dreieck                                    [820]
2170 CLS                                          [91]
2180 LOCATE 1,1:PRINT"Ausfuellen <"d$">":G [3026]
OSUB 90:IF fi=1 THEN 2210
2190 IF rx=-1 THEN d$="N"ELSE IF rx=1 THEN [1655]
 d$="J"
2200 GOTO 2180                                    [343]
2210 GOSUB 3580                                   [1005]
2220 GOSUB 190:IF f2=1 THEN 3700                  [1773]
2230 GOSUB 3720:GOSUB 3740:IF sp=1 THEN 22 [2912]
20 ELSE fl=1
2240 ax=x:ay=y                                    [872]
2250 MOVE ax,ay:DRAW x,y,fa:GOSUB 190:MOVE [6904]
 ax,ay:DRAW x2,y2,fa:GOSUB 3740:IF fi+f2=0
 THEN 2250
2260 IF f2=1 THEN 3700                            [1469]
2270 GOSUB 3720                                   [1025]
2280 bx=x:by=y                                    [897]
2290 MOVE ax,ay:DRAW x,y,fa:DRAW bx,by:DRA [7969]
W ax,ay:GOSUB 190:MOVE ax,ay:DRAW x2,y2,fa
:DRAW bx,by:DRAW ax,ay:GOSUB 3740:IF fi+f2
=0 THEN 2290
```

```
2300 IF f2=1 THEN 3700                            [1469]
2310 IF ax=bx AND ax=x AND bx=x THEN GOSUB [3606]
 3720:fl=0:GOTO 2220
2320 GOSUB 3720                                   [1025]
2330 PLOT-10,-10,zf:PRINT CHR$(23);CHR$(xo [1560]
);
2340 IF d$="J"THEN 2360                           [1129]
2350 MOVE ax,ay:DRAW bx,by:DRAW x,y:DRAW a [4356]
x,ay:PRINT CHR$(23);"1";:fl=0:GOTO 2220
2360 GOSUB 2370:ax=xp:ay=yp:x=xa:y=ya:bx=x [6541]
e:by=ye:GOSUB 2370:PRINT CHR$(23);"1";:fl=
0:GOTO 2220
2370 xa=ax:ya=ay:IF ABS(ax-bx)<ABS(ax-x)TH [6070]
EN xp=bx:yp=by:xe=x:ye=y ELSE xp=x:yp=y:xe
=bx:ye=by
2380 IF xa>xe THEN xv=xa:yv=ya:xa=xe:ya=ye [3069]
:xe=xv:ye=yv
2390 yd=ye-ya:st=xe-xa                            [1307]
2400 IF st>ABS(yd)THEN us=ab/4 ELSE us=(st [3143]
/ABS(yd))*ab/4
2410 FOR u=xa TO xe STEP us:MOVE xp,yp:DRA [4856]
W u,ya+(yd/st)*(u-xa):NEXT u
2420 RETURN                                       [555]
2430 '                                            [117]
2440 ' fuellen                                    [426]
2450 GOSUB 3580                                   [1005]
2460 GOSUB 190:IF f2=1 THEN 3700                  [1773]
2470 GOSUB 3740:IF sp=1 THEN 2460                 [2044]
2480 IF zf=0 THEN 190                             [787]
2490 PRINT CHR$(23);"0";:PLOT-10,-10,zf:ma [4672]
=640/ab-1:fx=INT(x/ab):fy=INT(y/2)
2500 IF TEST(x,y)=0 THEN|FILL,fx,fy,ab,ma         [1360]
2510 PRINT CHR$(23);"1";:GOTO 2460                [2524]
2520 '                                            [117]
2530 ' kopieren                                   [961]
2540 CLS                                          [91]
2550 LOCATE 1,1:PRINT"Hintergrund loeschen [5325]
 <"c$">":GOSUB 90:IF fi=1 THEN 2580
2560 IF rx=-1 THEN c$="N"ELSE IF rx=1 THEN [1768]
 c$="J"
2570 GOTO 2550                                    [365]
2580 GOSUB 3580                                   [1005]
2590 GOSUB 190:IF f2=1 THEN 3700                  [1773]
2600 ax=x:ay=y                                    [872]
2610 GOSUB 3720:GOSUB 3740:IF sp=1 THEN 25 [3090]
90 ELSE fl=1
2620 MOVE ax,ay:DRAW x,ay,fa:DRAW x,y:DRAW [8041]
 ax,y:DRAW ax,ay:GOSUB 190:MOVE ax,ay:DRAW
 x2,ay,fa:DRAW x2,y2:DRAW ax,y2:DRAW ax,ay
:GOSUB 3740:IF fi+f2=0 THEN 2620
2630 IF f2=1 THEN 3700                            [1469]
2640 xa=ax-x:IF xa<0 THEN sx=ab ELSE sx=-a [4081]
b
2650 ya=ay-y:IF ya<0 THEN sy=2 ELSE sy=-2         [2625]
2660 GOSUB 3720                                   [1025]
2670 MOVE x,y:DRAW x+xa,y,fa:DRAW x+xa,y+y [13248]
a:DRAW x,y+ya:DRAW x,y:GOSUB 190:MOVE x2,y
2:DRAW x2+xa,y2,fa:DRAW x2+xa,y2+ya:DRAW x
2,y2+ya:DRAW x2,y2:GOSUB 3740:IF fi+f2=0 T
HEN 2670
2680 GOSUB 3720                                   [1025]
2690 IF f2=1 THEN 3700                            [1469]
2700 PLOT-10,-10,zf:PRINT CHR$(23);"0";:OR [2754]
IGIN 0,0,0,639,16,399
2710 IF c$="J"THEN 2740                           [791]
2720 bx=x-ax+xa:by=y-ay+ya:FOR t=ax TO ax- [7046]
xa STEP sx:FOR i=ay TO ay-ya STEP sy:a=TES
T(t,i):IF a>0 THEN PLOT bx+t,by+i,a
2730 NEXT i,t:ORIGIN 0,0,0,639,0,399:PRINT [3178]
 CHR$(23);"1";:fl=0:GOTO 2590
2740 bx=x-ax+xa:by=y-ay+ya:FOR t=ax TO ax- [7697]
xa STEP sx:FOR i=ay TO ay-ya STEP sy:PLOT
bx+t,by+i,TEST(t,i):NEXT i,t
2750 ORIGIN 0,0,0,639,0,399:PRINT CHR$(23) [1760]
;"1";:fl=0:GOTO 2590
2760 '                                            [117]
2770 ' Zoom                                       [837]
2780 GOSUB 3580                                   [1005]
2790 fl=1:lx=32*ab:PRINT CHR$(23);"1";            [1186]
2800 MOVE x,y:DRAW x+lx,y,fa:DRAW x+lx,y+4 [8029]
8:DRAW x,y+48:DRAW x,y:GOSUB 190:MOVE x2,y
2:DRAW x2+lx,y2,fa:DRAW x2+lx,y2+48:DRAW x
2,y2+48:DRAW x2,y2:GOSUB 3740:IF fi+f2=0 T
HEN 2800
2810 IF f2=1 THEN 3700                            [1469]
2820 GOSUB 3720                                   [1025]
2830 FOR t=0 TO 31:FOR i=0 TO 23:zo(t,i)=T [3639]
EST(x+t*ab,y+i*2):NEXT i,t
2840 ax=x:ay=y                                    [872]
```

```
2850 |PUT:WINDOW#1,1,80/ab,1,24:CLS#1          [2106]
2860 PRINT CHR$(23);"0";                       [1485]
2870 FOR t=0 TO 31:FOR i=0 TO 23:f=zo(t,i)    [10025]
:IF f>0 THEN PLOT-10,-10,f:k=t*16:j=i*10:M
OVE k,16+j:DRAWR 11,0:MOVE k,18+j:DRAWR 11
,0:MOVE k,20+j:DRAWR 11,0:MOVE k,22+j:DRAW
R 11,0
2880 NEXT i,t                                  [449]
2890 FOR t=0 TO 8 STEP 8:PLOT 0,256+t,fa:D    [4261]
RAW 512+t,256+t:DRAW 512+t,16:NEXT t
2900 fl=0:PRINT CHR$(23);"1";                 [1843]
2910 GOSUB 190:IF f2=1 THEN 2950 ELSE IF s    [3031]
p=1 THEN 2910
2920 IF x>511 OR y>255 THEN 2910              [1115]
2930 px=INT(x/16):py=INT((y-16)/10):zo(px,    [6272]
py)=zf:PRINT CHR$(23);"0";:FOR i=0 TO 7 ST
EP 2:MOVE px*16,16+i+py*10:DRAWR 11,0,zf:N
EXT i
2940 PRINT CHR$(23);"1";:GOTO 2910            [2451]
2950 f2=0:PRINT CHR$(23);"0";:GOSUB 3580:P    [8231]
RINT CHR$(23);"0";:FOR t=0 TO 31:FOR i=0 T
O 23:PLOT ax+t*ab,ay+i*2,zo(t,i):NEXT i,t
2960 GOTO 2790                                 [317]
2970 '                                         [117]
2980 ' Text                                    [127]
2990 CLS:INPUT"Text eingeben:",te$            [2731]
3000 GOSUB 3580:fl=1:lt=LEN(te$)*8*ab+ab      [1470]
3010 PLOT-10,-10,fa:MOVE x-ab,y:DRAWR lt,0   [11907]
:DRAWR 0,20:DRAWR-lt,0:DRAWR 0,-20:GOSUB 1
90:PLOT-10,-10,fa:MOVE x2-ab,y2:DRAWR lt,0
:DRAWR 0,20:DRAWR-lt,0:DRAWR 0,-20:GOSUB 3
740:IF fi+f2=0 THEN 3010
3020 GOSUB 3720                               [1025]
3030 IF f2=1 THEN 3700                        [1469]
3040 PRINT CHR$(23);CHR$(xo);                 [2339]
3050 PLOT-10,-10,zf:MOVE x,y+16:TAG:PRINT     [3737]
te$;:TAGOFF:PRINT CHR$(23);"1";:GOTO 3010
3060 '                                         [117]
3070 ' Spray                                    [95]
3080 GOSUB 3580:pa=ab*4                       [1673]
3090 GOSUB 190:GOSUB 3740:IF sp+f2=0 THEN     [9400]
ORIGIN 0,0,0,639,16,399:PRINT CHR$(23);CHR
$(xo);:TAG:PLOT-10,-10,zf:MOVE x-pa,y+8:PR
INT CHR$(240);:TAGOFF:PRINT CHR$(23);"1";:
ORIGIN 0,0,0,639,0,399:GOTO 3090
3100 IF sp=1 THEN 3090 ELSE IF f2=1 THEN 3    [1875]
700
3110 '                                         [117]
3120 ' loeschen                                [951]
3130 l$="N":CLS                                [910]
3140 LOCATE 1,1:PRINT"Sind sie sicher <"l$    [5485]
"> ?":GOSUB 90:IF fi=1 THEN IF l$="N"THEN
RETURN ELSE 3170
3150 IF rx=-1 THEN l$="N"ELSE IF rx=1 THEN    [1441]
 l$="J"
3160 GOTO 3140                                 [453]
3170 CLS:|PUT:RETURN                           [633]
3180 '                                         [117]
3190 ' Bild speichern                         [2211]
3200 CLS                                        [91]
3210 GOSUB 3810                                [999]
3220 INPUT"Filename :",na$:PRINT:IF na$=""    [2614]
THEN RETURN
3230 IF dl$="J"THEN na$=LEFT$(na$,8):n1$=n   [11684]
a$+".bas":n2$=na$+".pic" ELSE IF fo$="Stan
dard" THEN na$=LEFT$(na$,12):n1$=na$:n2$="
!"+na$+".pic" ELSE n1$=na$:n2$=na$
3240 IF ba<2 THEN SPEED WRITE ba ELSE IF b    [6186]
a=2 THEN POKE &B8D1,2:POKE &B8D2,&1F ELSE
POKE &B8D1,0:POKE &B8D2,&15
3250 IF pa$="J" THEN GOSUB 3830               [1263]
3260 IF fo$="Standard" OR dl$="J" THEN SAV    [3019]
E n2$,b,&6000,&4000:RETURN
3270 n2$=LEFT$(n2$,64):v=&A000                [1660]
3280 FOR t=1 TO LEN(n2$):POKE v,ASC(MID$(n    [2986]
a$,t,1)):v=v+1:NEXT t
3290 FOR u=v TO &A03F:POKE u,0:NEXT u         [2028]
3300 PRINT:PRINT"Saving:";n2$                 [1590]
3310 CALL &BC6E:CALL &BC71:|HSAVE:|SAVE:RE    [4327]
TURN
3320 '                                         [117]
3330 ' Bild laden                             [1572]
3340 CLS                                        [91]
3350 GOSUB 3810                                [999]
3360 INPUT"Filename :",na$:PRINT:IF na$=""    [2980]
AND dl$="J"THEN RETURN
3370 IF na$=""THEN n2$="":GOTO 3390           [1677]
3380 IF dl$="J"THEN na$=LEFT$(na$,8):n1$=n   [10373]
a$+".bas":n2$=na$+".pic" ELSE IF fo$="Stan
dart" THEN na$=LEFT$(na$,12):n1$=na$:n2$=n
a$+".pic" ELSE n1$=na$:n2$=na$
3390 IF pa$="J" THEN GOSUB 3930               [1091]
3400 IF fo$="Standard" OR dl$="J" THEN LOA    [2105]
D n2$,&6000:RETURN
3410 n2$=LEFT$(n2$,64)                        [1181]
3420 PRINT"PLAY druecken"                     [2125]
3430 nf$="":|HLOAD:v=&A000                    [1704]
3440 b=PEEK(v):IF b>0 AND v<&A040 THEN nf$    [5270]
=nf$+CHR$(b):v=v+1:GOTO 3440
3450 IF UPPER$(nf$)<>UPPER$(n2$)AND n2$<>"    [3838]
"THEN PRINT"Found   :";nf$:GOTO 3430
3460 PRINT"Loading:";nf$:POKE &9FFF,255:|L    [2457]
OAD
3470 IF PEEK(&9FFF)<>255 THEN RETURN          [1975]
3480 PRINT:PRINT"Ladefehler !!!":PRINT CHR   [10067]
$(24)+"A"+CHR$(24)+"bbrechen    "+CHR$(24)+
"W"+CHR$(24)+"iederholen":PRINT:WHILE INKE
Y$<>"":WEND:a$=UPPER$(INKEY$)
3490 WHILE a$<>"A"AND a$<>"W":a$=UPPER$(IN    [3332]
KEY$):WEND
3500 IF a$="A"THEN RETURN                     [1391]
3510 GOTO 3420                                 [509]
3520 '                                         [117]
3530 ' Ende                                    [715]
3540 CLS:IF PEEK(&A300)=0 THEN PRINT"Basic    [3419]
 1.0" ELSE PRINT "Basic 1.1"
3550 PRINT:END                                 [738]
3560 :                                         [174]
3570 :                                         [174]
3580 MODE mo:FOR t=0 TO af:INK t,co(t):NEX    [2919]
T t:|GET
3590 x=320:y=200                               [986]
3600 fx=0:FOR t=1 TO af:FOR i=1 TO 25 STEP    [5284]
 ab:MOVE fx,0:DRAW fx,15,t:fx=fx+ab:NEXT i
,t
3610 BORDER co(zf):fa=1:FOR t=1 TO af:IF A    [4777]
BS(co(0)-co(t))>ABS(co(0)-co(fa))THEN fa=t
3620 NEXT t                                    [360]
3630 PEN fa:IF PEEK(&A300)=0 THEN ON mo GO    [9128]
SUB 3760,3770:LOCATE 16,25:PRINT "MENUE";:
ON mo GOSUB 3780,3790 ELSE LOCATE 16*2^mo,
25:PRINT "MENUE";
3640 LOCATE 1,1                                [611]
3650 |PUT                                      [413]
3660 ORIGIN 0,0,0,639,0,399                    [894]
3670 ug=(10/ge)                                [360]
3680 PRINT CHR$(23);"1";:fl=0:f2=0:RETURN     [2726]
3690 :                                         [174]
3700 LOCATE 1,25:PRINT SPACE$(20*(2^mo));:    [3931]
|PUT:RETURN
3710 :                                         [174]
3720 FOR t=1 TO 200:NEXT t:RETURN             [1163]
3730 FOR t=1 TO 10:NEXT t:RETURN              [2209]
3740 IF sp=0 THEN RETURN ELSE|GET:RETURN      [2876]
3750 :                                         [174]
3760 POKE &B1C8,0:POKE &B1CF,&CC:POKE &B1D    [2556]
0,&33:RETURN
3770 POKE &B1C8,0:POKE &B1CF,&F0:POKE &B1D    [2286]
0,&F:RETURN
3780 POKE &B1C8,1:POKE &B1CF,&88:POKE &B1D    [2496]
0,&44:RETURN
3790 POKE &B1C8,2:POKE &B1CF,&80:POKE &B1D    [2600]
0,&40:RETURN
3800 :                                         [174]
3810 IF dl$="J"THEN PRINT"Diskette einlege    [9917]
n und Taste druecken:":WHILE INKEY$="":WEN
D:PRINT:|DISC:|DIR,@pic$:PRINT:RETURN
3820 PRINT"Kassette einlegen":PRINT:RETURN    [4012]
3830 OPENOUT n1$                               [793]
3840 PRINT#9,"10 mode";STR$(mo)               [2150]
3850 PRINT#9,"20 for t=0 to";STR$(af);":re    [5329]
ad a:ink t,a:next t"
3860 PRINT#9,"30 data ";                      [1575]
3870 FOR t=0 TO af:PRINT#9,USING"##";co(t)    [3982]
;:IF t<af THEN PRINT#9,",";ELSE PRINT#9
3880 NEXT t                                    [360]
3890 PRINT#9,"40 load "+CHR$(34)+n2$+CHR$(    [3114]
34)+",49152"
3900 PRINT#9,"50 while inkey$="+CHR$(34)+C    [3916]
HR$(34)+":wend"
3910 CLOSEOUT                                  [902]
3920 RETURN                                    [555]
3930 OPENIN n1$:INPUT#9,a$:mo=VAL(RIGHT$(a    [2341]
$,3))
3940 LINE INPUT#9,a$:af=VAL(MID$(a$,15,2))    [3076]
:LINE INPUT#9,a$
3950 FOR t=0 TO af:co(t)=VAL(MID$(a$,9+t*3    [4270]
,2)):NEXT t:CLOSEIN:RETURN
```

Music Interrupt



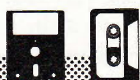
Wollten Sie schon lange einmal ein tolles Spiel programmieren, bei dem während des Spielablaufes eine fetzige Soundeinlage abgespielt wird?
Dann werden Sie schnell festgestellt haben, daß Ihr Programm durch diese Soundroutinen ziemlich an Geschwindigkeit verliert.
Erfahrungsgemäß muß man sich dann auf Kompromisse einlassen und das Programm von anderen zeitraubenden Routinen befreien. Das geht dann wiederum auf Kosten der Grafik oder der Präzision von Kollisionsabfragen.
Eine Alternative bietet das Programm Music Interrupt, mit dem ein simultaner Ablauf des Spielprogrammes und der Soundroutinen ohne nennenswerten Zeitverlust möglich ist.
Das Programm besteht aus einem kurzen Assemblerprogramm, mit dem der Interrupt eingehangen wird, sowie einem Datablock, der die Informationen über die abzuspielenden Töne enthält.
Eine genaue Beschreibung der Assemblerroutinen finden Sie im dokumentierten Sourcelisting.
Weiterhin haben Sie noch den Basic-Loader, der mit CALL &9000 die Musik abspielt und mit CALL &90B8 wieder Ruhe gibt.
Der Basic-Loader enthält schon einen Demo-Musikblock, der ab Adresse &906E steht. Diesen Musikblock können Sie gegen eigene Kompositionen austauschen, wobei folgende Bedingungen gelten:
Die Musikdaten werden fortlaufend in die Speicheradressen ab &906E geschrieben.
Die Parameter jeder Speicherstelle sind:
BYTE 0: Kanalangabe und Rendevouzerfordernisse
BYTE 1+2: Geräuschperiode (der eigentliche Ton)
BYTE 3+4: Länge des Tons
Diese Angaben sind gleichzeitig das Mindeste und Höchste, was vom Anwender eingegeben werden muß.
Die Auslesung erfolgt per Interrupt von der Treiberroutine. Die Daten für Hüllkurve usw. werden direkt im Treiber festgelegt.
Wer einen Assembler besitzt, sollte diesen zur Eingabe nehmen. Bei der Selbstprogrammierung der Musikstücke unter Basic erfordert es nämlich ganz schön viel Rechenarbeit, wenn man die benötigten 16 Bit-Werte in Low- und Highbyte zerlegen will.
Am Ende des Musikstückes muß der Wert 99 oder &63 stehen. Hiermit wird dem Treiber mitgeteilt, daß der Datenblock zu Ende ist und daß er mit dem Abspielen erneut beginnen soll.
Viel Spaß bei der "zeitunkritischen" Soundprogrammierung.

(M. Wogmann/TM)

```
1 MEMORY &8FFF                                [207]
2 :                                           [174]
3 dat=0:sz=0:dz=16                            [313]
4 FOR adr=&9000 TO &90E5                      [1068]
5 READ a$:b$="&":b$=b$+a$                     [1198]
6 byte=VAL(b$):POKE adr,byte                  [1068]
7 dat=dat+1                                   [116]
8 sz=sz+byte                                  [619]
9 IF dat <16 AND adr < &90E5  THEN 15         [1082]
10 READ check                                 [1299]
11 IF check <> sz THEN PRINT"FEHLER IN ZEI [3081]
LE ";dz
12 dz=dz+1                                    [721]
13 sz=0                                       [188]
14 dat=0                                      [239]
15 NEXT adr:CLEAR:GOTO 100                    [1697]
16 DATA 21,A1,90,7D,32,6B,90,7C,32,6C,90,C [2701]
D,41,90,3E,00, 1666
17 DATA 32,38,90,32,39,90,32,3A,90,32,3B,9 [3133]
0,32,3C,90,32, 1310
18 DATA 3D,90,32,3E,90,32,3F,90,32,40,90,2 [3720]
1,38,90,11,5F, 1321
19 DATA 90,0E,FF,06,81,CD,D7,BC,00,00,00,0 [2771]
0,00,00,00,00, 1156
20 DATA 00,3E,01,21,52,90,CD,BC,BC,3E,01,2 [3310]
1,59,90,CD,BF, 1628
21 DATA BC,C9,01,0F,FF,05,0F,01,0A,00,CE,0 [2926]
A,0A,32,0A,E5, 1206
22 DATA D5,C5,F5,DD,E5,FD,E5,2A,B6,90,C3,A [2599]
1,90,2A,B6,90, 2823
23 DATA 7E,32,DD,90,23,7E,32,E0,90,23,7E,3 [3434]
2,E1,90,23,7E, 1861
24 DATA 32,E4,90,23,7E,32,E5,90,21,DD,90,C [1907]
D,AA,BC,2A,B6, 2191
25 DATA 90,11,05,00,19,22,B6,90,7E,FE,63,C [2556]
A,AD,90,C3,D4, 1956
26 DATA 90,7E,CD,AD,BC,E6,07,CA,D4,90,C3,6 [3850]
D,90,21,E6,90, 2486
27 DATA 22,B6,90,C3,D4,90,E6,90,21,CA,90,7 [2267]
D,32,6B,90,7C, 2214
28 DATA 32,6C,90,21,E6,90,22,B6,90,C9,21,3 [3406]
8,90,CD,DD,BC, 2117
29 DATA C3,D4,90,C9,FD,E1,DD,E1,F1,C1,D1,E [1593]
1,C9,00,01,01, 2747
30 DATA 00,00,00,0F,00,00, 15                 [890]
100 CLEAR :REM * Dieser Teil ist der Noten [8021]
block, der im Pronzip nicht mehr mit abget
ippt werden brauch, der Anwender kann ab h
ier eigene Musikstuecke ablegen*
101 RESTORE 115                               [740]
102 dat=0:sz=0:dz=115                         [331]
103 FOR adr=&90E6 TO &93B6                    [705]
104 READ a$:b$="&":b$=b$+a$                   [1198]
105 byte=VAL(b$):POKE adr,byte                [1068]
106 dat=dat+1                                 [116]
107 sz=sz+byte                                [619]
108 IF dat<16 AND adr < &93B6 THEN 114        [1305]
109 READ check                                [1299]
110 IF check <> sz THEN PRINT"FEHLER IN ZE [3081]
ILE ";dz
111 dz=dz+1                                   [721]
112 sz=0                                      [188]
113 dat=0                                     [239]
114 NEXT adr                                  [547]
115 DATA 01,77,00,0A,00,02,7B,01,14,00,01, [2120]
71,00,0A,00,01, 401
116 DATA 7F,00,0A,00,02,9F,00,14,00,01,77, [3222]
00,0A,00,01,64, 549
117 DATA 00,0A,00,02,EF,00,14,00,01,71,00, [3153]
0A,00,01,77,00, 515
118 DATA 0A,00,02,0C,01,14,00,01,71,00,0A, [2811]
00,01,7F,00,0A, 307
119 DATA 00,02,9F,00,14,00,01,77,00,0A,00, [2658]
01,54,00,0A,00, 406
120 DATA 02,EF,00,14,00,01,50,00,0A,00,01, [1312]
77,00,0A,00,02, 484
121 DATA 66,01,14,00,01,71,00,0A,00,01,7F, [2722]
00,0A,00,02,96, 537
122 DATA 00,14,00,01,77,00,0A,00,01,64,00, [1845]
0A,00,02,EF,00, 502
123 DATA 14,00,01,71,00,0A,00,01,77,00,0A, [3207]
00,02,2D,01,14, 342
124 DATA 00,01,71,00,0A,00,01,7F,00,0A,00, [1921]
02,96,00,14,00, 434
125 DATA 01,77,00,0A,00,01,4B,00,0A,00,02, [2179]
EF,00,14,00,01, 478
126 DATA 59,00,0A,00,01,5F,00,0A,00,02,3F, [2403]
01,14,00,01,59, 381
127 DATA 00,0A,00,01,5F,00,0A,00,02,86,00, [3196]
14,00,01,64,00, 373
128 DATA 0A,00,01,5F,00,0A,00,02,BE,00,14, [1598]
00,01,50,00,0A, 419
129 DATA 00,01,43,00,0A,00,02,E1,00,14,00, [2743]
01,3C,00,0A,00, 396
130 DATA 01,43,00,0A,00,02,86,00,14,00,01, [2534]
47,00,0A,00,01, 317
131 DATA 43,00,0A,00,02,BE,00,14,00,01,38, [2095]
00,0A,00,01,3C, 417
```

```
132 DATA 00,0A,00,02,2D,01,14,00,01,38,00, [3080]
0A,00,01,3C,00, 206
133 DATA 0A,00,02,96,00,14,00,01,3F,00,0A, [2980]
00,01,3C,00,0A, 327
134 DATA 00,02,B3,00,14,00,01,50,00,0A,00, [2123]
01,4B,00,0A,00, 378
135 DATA 02,EF,00,14,00,01,43,00,0A,00,01, [1815]
4B,00,0A,00,02, 427
136 DATA 96,00,14,00,01,50,00,0A,00,01,4B, [2376]
00,0A,00,02,B3, 528
137 DATA 00,14,00,01,5F,00,0A,00,01,59,00, [2983]
0A,00,02,18,02, 254
138 DATA 14,00,01,50,00,0A,00,01,59,00,0A, [2389]
00,02,E1,00,14, 458
139 DATA 00,01,5F,00,0A,00,01,59,00,0A,00, [3216]
02,3F,01,14,00, 292
140 DATA 01,77,00,0A,00,01,71,00,0A,00,02, [2393]
66,01,14,00,01, 380
141 DATA 64,00,0A,00,01,71,00,0A,00,02,E1, [2487]
00,14,00,01,77, 601
142 DATA 00,0A,00,01,71,00,0A,00,02,3F,01, [2416]
14,00,01,7F,00, 348
143 DATA 0A,00,01,77,00,0A,00,02,DE,01,14, [2617]
00,01,71,00,0A, 509
144 DATA 00,01,77,00,0A,00,02,EF,00,14,00, [3533]
01,7F,00,0A,00, 529
145 DATA 01,77,00,0A,00,02,2D,01,14,00,01, [1968]
9F,00,0A,00,01, 369
146 DATA 96,00,0A,00,02,66,01,14,00,01,86, [1897]
00,0A,00,01,96, 581
147 DATA 00,0A,00,02,EF,00,14,00,01,9F,00, [2881]
0A,00,01,96,00, 592
148 DATA 0A,00,02,2D,01,14,00,01,B3,00,0A, [2778]
00,01,BE,00,0A, 469
149 DATA 00,02,DE,01,14,00,01,9F,00,0A,00, [2714]
01,86,00,0A,00, 560
150 DATA 02,BE,00,14,00,01,71,00,0A,00,01, [2379]
5F,00,0A,00,02, 444
151 DATA 0C,01,14,00,01,50,00,0A,00,01,43, [2526]
00,0A,00,02,3F, 267
152 DATA 01,14,00,01,38,00,0A,00,01,3C,00, [3367]
0A,00,02,BE,00, 351
153 DATA 14,00,01,43,00,0A,00,01,4B,00,0A, [3242]
00,02,0C,01,14, 219
154 DATA 00,01,50,00,0A,00,01,43,00,0A,00, [2269]
02,66,01,14,00, 294
155 DATA 01,3C,00,0A,00,01,4B,00,0A,00,02, [2565]
B3,00,14,00,01, 359
156 DATA 50,00,0A,00,01,4B,00,0A,00,02,EF, [2882]
00,14,00,01,59, 527
157 DATA 00,0A,00,01,5F,00,0A,00,02,2D,01, [2489]
14,00,01,50,00, 265
158 DATA 0A,00,01,59,00,0A,00,02,B3,00,14, [2012]
00,01,5F,00,0A, 417
159 DATA 00,01,59,00,0A,00,02,E1,00,14,00, [1680]
01,71,00,0A,00, 471
160 DATA 63, 99                             [397]
```

CPC-Monitor



Der CPC-Monitor ist ein komfortabler Maschinensprache-Monitor, der zu 100 % in Maschinensprache ist, und nur etwas mehr als 7 KByte belegt.
Er beinhaltet einen Assembler, Disassambler (mit der Möglichkeit zur Ausgabe auf dem Drucker), Monitor-Teil, eine Verschiebroutine (in welcher Sprünge in den zu verschiebenden Block automatisch auf die neue Adreesse angepaßt werden), eine Breakpoint-Option, sowie noch einige zusätzliche Befehle, die das Programmieren erleichtern. Eine genaue Erklärung der Befehle folgt in der Anleitung.

Der CPC-Monitor beginnt an der Adresse &8EB0 und endet an Adresse &AB0F; die Adressen &AB10 - &AB7F werden zum Abspeichern von Flags und Registerinhalten benutzt. Eine genauere Speicherbelegung finden Sie im Anhang.

Anleitung

1. Eingabe und Sicherung des Programms
Nachdem Sie das Basic-Listing des CPC-Monitors eingegeben und gestartet haben, werden die Werte in den Data-Zeilen ab Adresse &8EB0 in den Speicher gePOKEd. Sollten Sie beim Abtippen einen Fehler gemacht haben, so bekommen Sie die Zeilennummer, in der der Fehler auftrat, ausgedruckt und brauchen nur diese Zeile auf den Fehler hin zu untersuchen.
Allerdings sollten Sie beim Eintippen der Prüfsumme am Ende jeder Zeile besonders aufpassen, da es sonst zu unerwarteten Fehlern kommen kann.
Sind nun alle Data-Zeilen fehlerfrei, so sollten Sie zunächst einmal das Basic-Listing mit <SAVE"CPC-MON-DATAS"> und anschließend das Maschinenprogramm mit <SAVE"CPC-MON",b,&8EB0,&AB0F -&8EB0> abspeichern. Nun können Sie den CPC-Monitor mit CALL &8EB0 starten.
Der CPC-Monitor wurde auf einem Schneider CPC 464 geschrieben; da der CPC 664 und der CPC 6128 ein erweitertes 464-RAM haben, müßte das Programm auch auf ihnen laufen.

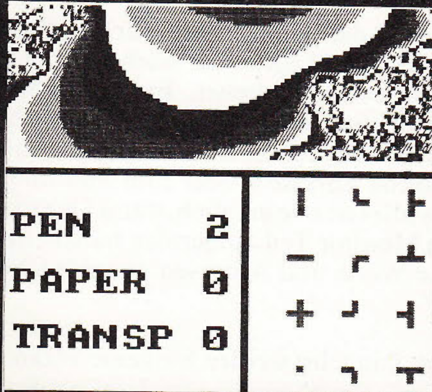
2. Laden und Starten des Programms
Wenn Sie dann wieder einmal den CPC-Monitor laden wollen, so müssen Sie <MEMORY &8EAF> eingeben, um das Programm vor dem Überschreiben durch das Betriebssystem zu schützen.
Danach geben Sie <LOAD"CPC-MON">. ein. Ist dann der CPC-Monitor nach dem Ladevorgang im Speicher, können Sie ihn mit <CALL &8EB0> starten oder Ihr selbsterstelltes Maschinenprogramm noch in den Speicher laden, bevor Sie den CPC-Monitor aufrufen. Nach dem Starten befinden Sie sich in der Menueebene des CPC-Monitors und können mit der Programmierung beginnen.
Doch zuvor noch eine Beschreibung des Bildschirmaufbaus.

3. Der Bildschirmaufbau des CPC-Monitor
In der obersten Zeile steht der Programmname, sowie mein Name. In der untersten Zeile finden Sie die Kürzel der wichtigsten Befehle. Am rechten Bildrand sind die Inhalte der Z80-Register aufgeführt. Diese haben nach dem Aufruf des Monitors zufällige Werte ohne jegliche Aussagekraft. Sie werden sich sicherlich fragen, wieso sie dann ausgedruckt werden. Das hat folgenden Grund: Sie dienen der Fehlersuche in Ihrem erstellten Programm, nachdem Sie es vom CPC-Monitor aus, aufgerufen haben. Denn dann werden die letzten dem Register zugeschriebenen Werte ausgedruckt. Hat dann ein Register nicht den erhofften Wert, so wurde es falsch benutzt. Die zwei Register sind invers dargestellt.
Des weiteren gibt es noch zwei Fenster. Das erste und wichtigste liegt in der linken Bildschirmhälfte und dient zum Assemblieren, Disassemblieren, Verschieben, usw. Das zweite liegt in der rechten Bildschirmhälfte und reicht bis zum Registerausdruck. Es dient ausschließlich zum Betrieb des Monitorteils - doch davon später.

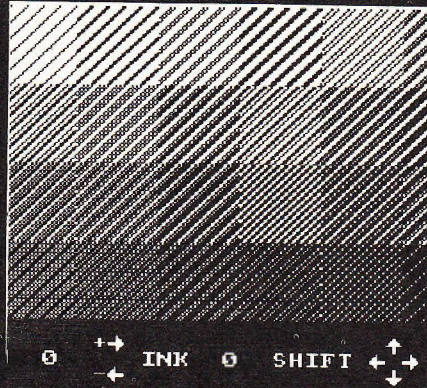
4. Programmierbefehle allgemein...
sind beim CPC-Monitor nicht ganze Wörter, wie etwa beim Basic, sondern Kürzel von einem Zeichen (Buchstaben) Länge, die einen Befehl symbolisieren. Bei einigen Befehlen werden noch Adressen oder Werte verlangt, welche Sie dezimal oder hexadezimal eingeben können. Es spielt auch keine Rolle, ob Sie Groß- oder Kleinbuchstaben benutzen.

DMV präsentiert

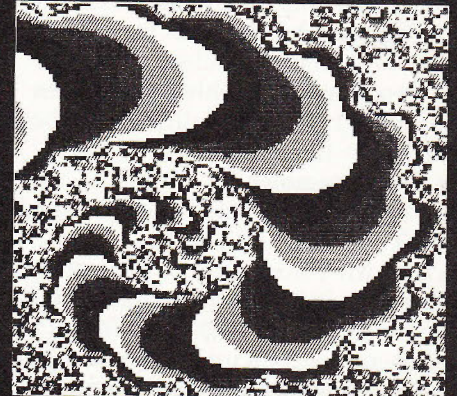
Das universelle Hardcopy-Programm für Schneider CPC 464/664/6128



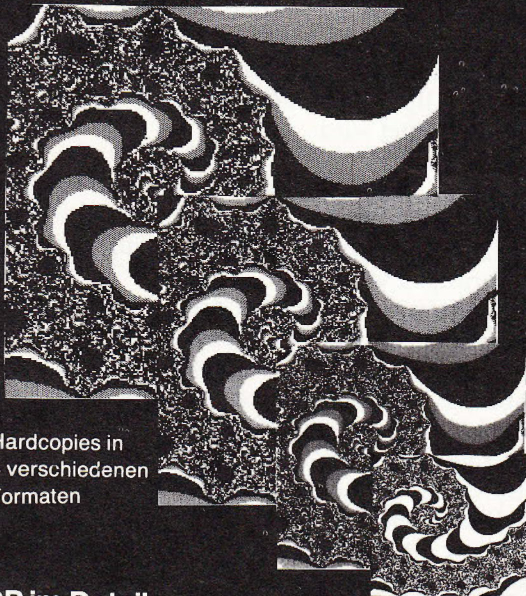
...komfortable Menuesteuerung für Bildgestaltung



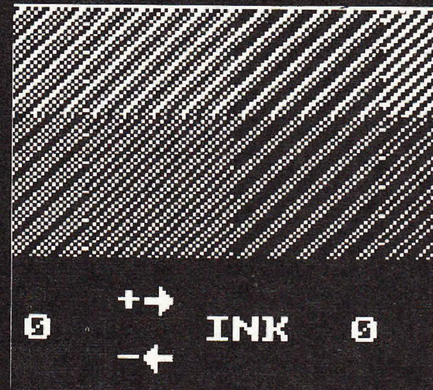
...und Farben/Raster-Auswahl



...Hardcopy-Simulation auf den Bildschirm



...Hardcopies in 4 verschiedenen Formaten



...beliebige Ausschnitts-vergrößerungen

COPYSHOP im Detail:

- Hardcopy in 4 (!) Formaten: DIN A4, DIN A5, 13,5 x 8,5 cm und 21,5 x 13,5 cm
- superschnelle Hardcopy-Routine: DIN A4 in ca. 4 Minuten
- arbeitet in allen 3 Modes
- Anpaßmenue für JEDEN Matrixdrucker
- 32 Farbraster über Menue wählbar
- Grafikeditor
- komfortable Pull-Down-Menues
- schnelle Fill-Routine
- beliebige Ausschnittvergrößerungen
- Bildschirm invertieren
- selbstrelozierbare Hardcopy-Routinen für eigene Programme
- neue Save- und Load-Routinen erkennen automatisch Mode und Farbwerte
- Freezer – saved auf Tastendruck Screenshots aus laufenden Programmen, die anschließend ausgedruckt werden können

Und die Weltneuheit: **Hardcopy-Simulator auf dem Bildschirm!!**
Sie können sich Ihre Hardcopy vor dem endgültigen Ausdruck auf dem Bildschirm ansehen!

COPYSHOP ist das ultimative Hardcopy-Programm für alle Schneider Computer.
Erhältlich auf Kassette (DM 59,–) und 3"-Diskette (DM 69,–)
inkl. ausführlicher Bedienungsanleitung.

COPYSHOP gibt es im guten Fachhandel oder direkt bei:

DMV-Verlag, Fuldaer Straße 6, 3440 Eschwege

BEI DIREKTZUSTELLUNG ZZGL. DM 3,– PORTO/VERPACKUNG; PER NACHNAHME ZZGL. NACHNAHMEGEBÜHR.
IN DAS AUSLAND IST NACHNAHME NICHT MÖGLICH.

Händleranfragen erwünscht!

Falls der Befehl noch Adressen oder Werte verlangt, so müssen Sie nach dem Kürzel noch ein Zeichen unterbringen (z.B. Leerzeichen oder Punkt). Um einen Befehl in der Menueebene abzuschließen, drücken Sie <ENTER>. Sollte Ihre Eingabe nicht korrekt sein, so können Sie sie mit <CLR> und <DEL> bearbeiten oder einfach überschreiben, indem Sie den Cursor mittels der Pfeiltasten auf die gewünschte Position innerhalb der Zeile bringen und dann die neuen Zeichen eingeben.

5. Programmierbefehle im einzelnen

Nun folgt eine Aufstellung aller beim CPC-Monitor verfügbaren Befehle:

a. Der Assembler...

wird mit <A> und einer Adresse aufgerufen: Bsp: A 1000.

Dann wird diese Adresse in hexadezimaler Form ausgegeben und ein Maschinensprachebefehl erwartet. Ist der eingegebene Befehl nicht korrekt, so wird die Adresse nochmals ausgegeben und es wird ein neuer Maschinenbefehl oder ein <S> (um zur Menueebene zurückzukehren) erwartet.

Selbstverständlich sind alle Z80-Befehle uneingeschränkt benutzbar.

Ist der Befehl korrekt, so wird er übersetzt und die Werte (bzw. der Wert) ab der angegebenen Adresse im Speicher abgelegt. Danach wird die Adresse um die entsprechende Anzahl der Werte erhöht und in der nächsten Zeile ausgegeben. Die Bearbeitungszeit ist kaum feststellbar!

Noch eine Anmerkung zum JR: Sie können den Offset entweder als Distanz (JR $+100/ JR $-10) oder als Adresse (JR &9000) eingeben. Haben Sie eine Adresse eingegeben, so wird die Distanz aus dieser und der aktuellen Adresse errechnet. Die Distanz darf nur zwischen -126 und + 129 liegen.

Außerdem stehen Ihnen noch zwei zusätzliche Befehle zur Verfügung:

1.<W:wort:>

übergibt eine Zeichenkette (deren ASCII-Werte)

2.<Z:12345>

übergibt eine bis zu 16 Bit hohe Zahl.

Bei beiden Befehlen muß ein Doppelpunkt nach dem Kürzel stehen.

b. Der Disassembler

Wird auf drei Arten mit <D> und einer oder zwei Adresse aufgerufen.

1. D adr:

Der Befehl, der an dieser Adresse steht, wird übersetzt und danach wird in die Menueebene zurückgesprungen.

2. D adr-:

Ab der angegebenen Adresse werden zehn Befehle übersetzt, danach wird eine Eingabe von Ihnen erwartet. Ist diese Eingabe ≠ <S>, wird der Vorgang wiederholt, ansonsten wird in die Menueebene gesprungen.

3. D anfadr.-endadr.:

Funktioniert wie 2., wird jedoch die endadr. erreicht, so wird automatisch abgebrochen.

Falls nun ein Befehl 4 Bytes belegt (Adresse 1000 bis 1003) und Sie D 1002 eingegeben haben, so bekommen Sie trotzdem den richtigen Ausdruck, da der CPC-Monitor jede Adresse, die nicht auf das erste Byte eines Maschinenbefehls zeigt, auf das erste Byte anpaßt.

c. Der Drucker...

wird wie der Disassembler auf drei Arten mit <P> statt <D> aufgerufen (<P> weil Printer). Da Sie die Printer-Routine nicht mit <S> abbrechen können, ist die zweite Art nicht empfehlenswert.

d. Der Monitor-Teil...

kann wie der Disassembler auf drei Arten mit <M> statt <D> aufgerufen werden. Jedoch wird nach Beendigung des Ausdrucks nicht in die Menueebene zurückgesprungen, es können vielmehr Speicherinhalte, die im Monitor-Fenster stehen, verändert werden. Sie tun dies, indem Sie den Cursor mittels der vier Pfeiltasten über die zu verändernde Zahl bringen und den neuen Wert einfach darüberschreiben.

Befindet sich der Cursor in den zuletzt ausgedruckten Zeilen, können Sie mittels <ENTER> die nächsten acht Werte ausdrucken lassen und bearbeiten. Erst wenn Sie sich im Monitor-Fenster befinden, gelangen Sie durch <S> wieder zurück in die Menueebene. Von dort aus können Sie durch <M> ohne Adresse wieder zum Monitor-Fenster zurückkehren (dies ist nur möglich, wenn Sie vorher schon einmal den Monitor-Teil aufgerufen haben, da ansonsten noch keine Werte und Adressen ausgedruckt werden).

e.) Umschalten

<U> eingeben genügt. Zunächst werden Sie keine Veränderung bemerken.

Wenn Sie aber den Monitor-Teil aufrufen, werden Sie feststellen, daß an die Stelle der Registerinhalte nun die entsprechenden ASC-Zeichen der ausgelesenen Werte gedruckt werden. Sie sollten deswegen erst <U> drücken, wenn Sie die Registerinhalte nicht mehr brauchen (Werte kleiner als 32 werden als Punkt ausgedruckt).

Nochmaliges Eingeben von <U> bewirkt, daß wieder umgeschaltet wird.

f. Run

Ein von Ihnen geschriebenes Programm kann mittels <R> und einer Adresse aufgerufen werden (R 2000).

Ihr Programm muß mit RET enden, damit richtig zurückgesprungen wird.

Sie sollten besonders auf die Richtigkeit des Stack Pointers (SP = Stapelzeiger) achten, denn wenn Sie ein PUSH oder ein POP zuviel ausführen, kommt es sehr leicht zum Absturz. Um dem zu entgehen, sollten Sie anfangs die Rücksprungadresse retten [mit POP HL/LD(Adresse),HL] und vor dem Rücksprung wieder auf den Stapel legen [durch: LD HL,(Adresse)/PUSH HL].

g. Der Breakpoint

Um die Fehlersuche in Ihrem Programm zu erleichtern, können Sie an eine beliebige Stelle Ihres Programms einen Breakpoint (sinngemäß: Kontrollpunkt) setzen, indem Sie <K> und eine Adresse eingeben. Ab dieser Adresse steht dann ein Sprung zum CPC-Monitor. In diesem Zusammenhang ist der Ausdruck der Registerinhalte bei der Fehlersuche sehr nützlich

Hinweis: Der Breakpoint belegt 3 Bytes, so daß Sie darauf achten sollten, daß der Breakpoint nicht wichtige Befehle überschreibt, deren Fehlen beim Ansprung zu Fehlergebnissen führen würden. Den Breakpoint können Sie durch <K 0> löschen. Die drei durch den Breakpoint ersetzen Bytes werden zurückgegeben.

Auch durch die Eingabe eines neuen Breakpoints wird der alte gelöscht, ebenso wie es auch beim Rücksprung in Basic geschieht, damit Sie Ihr Programm ohne Breakpoint abspeichern können.

h. Basic

Durch <B> kehren Sie ins Basic zurück und können Ihr geschriebenes Programm mit <SAVE"name",B,anfadr., länge> abspeichern oder an einem Basicprogramm schreiben.

i. Füllen

Füllt einen Block mit einem Wert → z.B. F 1000,2000,32 füllt die Speicherzellen von 1000 - 2000 mit 32.

Allgemein: F anfadr.,endadr, wert.

j. Verschieben
Verschiebt einen Block an eine neue Adresse. Wird mit <V> und drei durch Komma getrennte Adresse aufgerufen:
V anfadr , endadr., neüadr.. anfadr. und endadr. legen den zu verschiebenden Block fest, während neueadr. den neuen Blockanfang festlegt.

k. Eingeben
Dient zur Unterstützung der Verschieb-Routine. Falls Sie einen Block an eine neue Adresse verschoben haben und diesen Block von Ihrem Programm aus anspringen wollen (mit JP oder Call) müßten Sie alle Sprungadressen, die in diesen Block zeigen, anpassen. Genau dies wird von dieser Routine übernommen. Und zwar geschieht dies folgendermaßen:
Sie müssen dem CPC-Monitor vor Benutzung der Verschieb-Routine einen Block angeben, welcher dann beim Aufruf der Verschieb-Routine auf die o.g. Sprünge durchsucht wird, und diese anpaßt. Dies wird durch <E> anfadr., endadr. bewerkstelligt (z.B.: E &8000,&9000). Diese Adressen müssen Sie nur einmal eingeben. Haben Sie vor dem ersten Aufruf der Verschieb-Routine die Adressen nicht eingegeben, so wird nur verschoben. Um diesen Befehl abzustellen, geben Sie <E 0,0> ein. Wollen Sie den Bereich ändern, dann geben Sie einen neuen E-Befehl ein.

l. Suchen
Sucht einen angegebenen Wert innerhalb eines anzugebenden Blocks und druckt die Adresse(n) in denen der Wert gefunden wurde. Wird durch <S anfadr., endadr., wert> aufgerufen (<S 1000,2000,10>.
Der Wert sollte 255 nicht überschreiten. Sie können die Ausgabe durch eine beliebige Taste abbrechen.

m. Cls
Löscht das linke Fenster.<C> eingeben genügt.

n. Zusatz
Mit <Z> können Sie zwischen RAM und dem ROM in &0000 - &3FFF und &C000 - &FFFF umschalten. Somit können Sie sich z.B. mit dem Disassembler das ROM anschauen. Um ins RAM zurückzuschalten, drücken Sie erneut <Z>.

Anhang:

Adresse	Inhalt
8EB0 - 8F07	Verschiedenes
8F08 - 8F1E	Drucker-Routinen
8F1F - 8F84	Breakpoint
8F85 - 8F92	Befehl interpretieren 3
8F93 - 9002	Verschiedenes
9003 - 9020	Wort-Befehl beim Assembler
9021 - 903E	Zahl-Befehl beim Assembler
903F - 9081	Befehl interpretieren 2
9082 - 90A9	<DEL> ausführen
90AA - 90CE	<CLEAR> ausführen
90CF - 9102	Disassembler: Adresse holen
9103 - 9135	Disassembler: Hauptschleife
9136 - 9377	Disassembler: UPRO's
9378 - 97AC	Disassembler: interpretieren
97AD - 97ED	Monitor: Adresse holen
97EF - 9867	Monitor: Werte holen und drucken
9868 - 9979	Monitor: Speicherinhalte ändern
997A - 99B2	Fill
99B3 - 99DB	Cls
99DC - 9A45	Run
9A46 - 9A9B	verschieben
9A9C - 9AEF	Verschieben - Sprunganpassung
9AF0 - 9BFF	UPRO's
9C00 - 9CBF	Zeichenketten einiger Z80-Befehle für den Disassembler
9CC0 - 9D68	Assembler: übersetzten Befehl in Speicher legen
9D69 - 9EEB	UPRO's
9DE6 - 9E8F	Eingegebene Zeichenkette in abarbeitbare Zahl oder Adresse wandeln
9EEC - A047	Assembler: Interpretation des Operanden
A048 - A268	Zeichenketten zum Vergleichen und Ausdrucken
A269 - A394	Programmstart: Initialisierung und Bildschirmaufbau
A395 - A41A	Tastaturabfrage bzw. Befehlseingabe
A41B - A479	Cursorposition auf nächste Abfrage vorbereiten
A47A - A4B1	Assembler: aktivieren und aktuelle Adresse ausgeben
A4B2 - AAFF	Assembler: Befehlsinterpretation
A4EB - A57A	Assembler: Sprungtabelle zu Befehlen mit Operand
A57B - A5D1	Assembler: Befehle ohne Operand
A5D2 - AAFF	Assembler: Interpretation eines Befehls mit Operand
AB00 - AB0F	Befehl interpretieren 1

```
10 DATA C3,69,A2,3A,37,AB,F5,CD,C0,8F,F1,3 [4834]
2,37,AB,C9,E1,C3,C3,91,ED, 3246
11 DATA 43,46,AB,ED,5B,3F,AB,D5,C3,87,9B,3 [3160]
A,37,AB,FE,FF,C2,1B,A4,3E, 2808
12 DATA 0D,CD,16,8F,3E,0A,CD,16,8F,C9,ED,4 [5300]
3,85,B2,CD,7A,9D,7D,CD,7A, 2577
13 DATA 9D,CD,1B,A4,CD,1B,BB,FE,01,30,06,F [3022]
1,C1,E1,C3,BA,8F,E1,E1,C3, 3109
14 DATA 95,A3,CD,7A,99,C3,8E,99,3E,FF,32,3 [3673]
7,AB,C3,D4,90,FE,FF,C0,3A, 3185
15 DATA 35,AB,CD,2E,BD,38,FB,CD,31,BD,C9,3 [3778]
E,29,32,15,98,AF,32,8F,9A, 2463
16 DATA 3A,D9,91,FE,00,C8,AF,32,D9,91,ED,5 [1750]
B,EB,8F,21,DA,91,01,03,00, 2567
17 DATA ED,B0,C9,21,6D,AB,CD,87,9B,21,00,0 [2371]
0,B7,ED,42,3A,D9,91,38,0B, 2428
18 DATA FE,00,CA,61,A4,CD,2E,8F,C3,95,A3,F [3690]
E,00,28,05,C5,CD,2E,8F,C1, 2701
19 DATA 3C,32,D9,91,ED,43,EB,8F,60,69,11,D [3114]
A,91,00,01,03,00,C5,ED,B0, 2349
20 DATA ED,5B,EB,8F,C1,21,ED,8F,ED,B0,C3,9 [2985]
5,A3,FE,4B,28,B6,FE,50,CA, 3319
21 DATA 08,8F,FE,53,C2,61,A4,CD,7A,99,79,2 [2991]
A,46,AB,ED,5B,44,AB,B7,ED, 2814
22 DATA 52,18,13,E1,C3,61,A4,3E,01,32,3A,A [3709]
B,C9,CD,C0,8F,3E,0B,32,37, 2067
23 DATA AB,C9,23,44,4D,EB,ED,B1,E5,C3,F0,9 [3304]
A,11,0A,00,ED,52,22,51,AB, 2651
24 DATA 7B,32,37,AB,E5,CD,78,93,D1,2A,51,A [3895]
B,ED,4B,44,AB,B7,ED,42,2A, 2682
25 DATA 51,AB,28,06,38,EA,ED,53,51,AB,AF,3 [3746]
2,37,AB,C9,D0,07,C3,EF,99, 2614
26 DATA AF,32,37,AB,C3,95,A3,3A,37,AB,FE,0 [3573]
0,CA,18,BB,C3,1E,BB,00,23, 2356
27 DATA 7E,FE,3A,C2,E0,9E,06,0D,ED,5B,5E,A [1895]
B,23,7E,FE,3A,28,04,12,13, 2180
28 DATA 10,F6,42,4B,00,00,C3,89,A4,23,7E,F [2795]
E,3A,C2,E0,9E,23,E5,CD,E6, 2647
29 DATA 9D,E1,2A,5E,AB,71,23,AF,B8,28,02,7 [3199]
0,23,44,4D,00,C3,89,A4,FE, 2280
30 DATA 01,CA,B2,A4,3A,6B,AB,FE,41,CA,7A,A [4776]
4,FE,42,CA,1F,8F,00,FE,44, 2706
31 DATA CA,D4,90,FE,4D,CA,AD,97,FE,46,CA,0 [4036]
2,8F,FE,56,CA,46,9A,FE,52, 3188
32 DATA CA,DC,99,FE,43,CA,B3,99,FE,45,CA,A [3629]
3,9B,FE,55,CA,2C,9B,FE,5A, 3357
33 DATA CA,A2,97,C3,85,8F,3A,60,AB,FE,01,2 [2835]
8,0A,3A,5B,AB,FE,00,CA,AE, 2566
34 DATA A3,18,08,3A,60,AB,FE,05,CA,AE,A3,3 [3414]
D,32,5B,AB,21,57,AB,35,2E, 2081
35 DATA 5C,35,21,86,B2,35,2A,85,B2,E5,2A,5 [2905]
C,AB,3E,7F,95,47,54,5D,13, 2035
36 DATA 1A,77,00,CD,5A,BB,00,13,23,10,F5,3 [3772]
E,20,CD,5A,BB,E1,22,85,B2, 2088
37 DATA C3,AE,A3,C8,E1,C3,C3,91,21,6D,AB,C [4623]
D,87,9B,2A,3D,AB,ED,43,44, 2946
```

```
38 DATA AB,7E,FE,20,28,11,FE,2D,C2,61,A4,2 [3420]
3,7E,01,FF,FF,FE,20,28,03, 2395
39 DATA CD,87,9B,2A,44,AB,ED,43,53,AB,CD,B [4367]
3,8E,06,0A,ED,5B,56,AB,ED, 2698
40 DATA 53,85,B2,C5,2A,51,AB,7C,CD,7A,9D,7 [2626]
D,CD,7A,9D,00,CD,44,9B,CD, 2735
41 DATA 78,93,CD,CF,8E,21,52,AB,3A,54,AB,C [2750]
1,BE,DA,F0,8F,3A,53,AB,2B, 2759
42 DATA BE,DA,F0,8F,10,CD,CD,F7,8F,06,0A,C [3686]
D,E0,9D,FE,53,20,C1,C3,F0, 3206
43 DATA 8F,32,35,AB,3A,37,AB,FE,00,C2,10,8 [3751]
F,3A,35,AB,C3,5A,BB,CD,CB, 2470
44 DATA 91,CD,6A,93,E6,07,18,0B,CD,CB,91,C [3412]
D,6A,93,E6,38,0F,0F,0F,FE, 2471
45 DATA 06,47,20,17,3A,39,AB,FE,00,78,20,0 [4620]
8,3A,3C,AB,FE,00,C2,97,92, 1866
46 DATA 21,35,A2,CD,9E,91,C9,3E,07,90,21,F [4449]
7,A1,85,6F,06,01,18,1C,06, 1920
47 DATA 02,18,18,21,73,A1,06,03,18,11,06,0 [2989]
4,18,0D,06,04,CD,AF,91,3E, 1053
48 DATA 20,CD,45,91,C9,06,09,3A,37,AB,CD,4 [4551]
9,A2,00,7E,FE,2E,20,02,D6, 2065
49 DATA 0E,23,CD,45,91,10,F3,2A,51,AB,23,2 [3403]
2,51,AB,C9,2A,51,AB,2B,22, 1914
50 DATA 51,AB,C9,32,8F,9A,C3,95,A3,00,C9,0 [2056]
0,00,00,3E,26,CD,45,91,79, 2148
51 DATA CD,7A,9D,C9,21,2F,9C,CD,93,91,CD,C [2504]
B,91,CD,6A,93,CD,7A,9D,18, 2937
52 DATA CA,3E,2C,CD,45,91,CD,C3,91,3E,26,C [3740]
D,45,91,CD,6A,93,57,2B,CD, 2584
53 DATA 6A,93,5F,7A,CD,7A,9D,7B,CD,7A,9D,1 [4400]
8,AA,CD,27,92,3E,2C,CD,45, 2525
54 DATA 91,C9,3E,2C,CD,45,91,3E,28,CD,45,9 [4051]
1,CD,FE,91,3E,29,CD,45,91, 2518
55 DATA C9,2A,51,AB,CD,6A,93,CD,C3,91,16,0 [3126]
0,FE,80,30,0F,5F,19,3E,26, 2185
56 DATA CD,45,91,7C,CD,7A,9D,7D,C3,7A,9D,E [3564]
D,44,5F,B7,ED,52,18,EB,CB, 2990
57 DATA 3F,CB,3F,CB,3F,07,07,85,6F,C9,CD,C [2883]
B,91,CD,6A,93,E6,38,0F,0F, 2381
58 DATA 0F,47,3E,08,90,47,21,0D,A2,07,85,6 [3092]
F,78,FE,03,DA,8F,91,FE,05, 1972
59 DATA DA,93,91,CB,40,CA,93,91,C3,8F,91,3 [4405]
E,05,BD,C2,93,91,18,35,21, 2606
60 DATA 41,A2,FE,FD,28,03,21,45,A2,CD,9E,9 [2362]
1,3A,38,AB,FE,00,20,06,CD, 2331
61 DATA CB,91,CD,CB,91,CD,6A,93,4F,CD,DE,9 [3957]
1,3E,29,CD,45,91,3A,38,AB, 2817
62 DATA FE,00,CC,C3,91,CD,C3,91,C9,CD,CB,9 [3442]
1,21,05,A2,3A,3C,AB,FE,00, 2840
63 DATA 28,0A,21,0B,A2,FE,FD,28,03,21,0D,A [3763]
2,C3,93,91,3A,3C,AB,FE,00, 2044
64 DATA C8,E1,E1,18,DF,CD,F3,92,C3,93,91,C [5129]
D,CB,91,CD,6A,93,E6,30,0F, 3282
65 DATA 0F,0F,0F,47,3E,04,90,32,3B,AB,FE,0 [3352]
2,CC,E3,92,3A,3B,AB,21,01, 1761
66 DATA A2,07,85,6F,C9,BE,C8,23,10,FB,05,C [3254]
9,CD,AF,91,3A,3C,AB,FE,00, 2580
67 DATA C8,41,21,86,B2,35,10,FD,18,9B,3A,3 [3233]
C,AB,FE,00,C4,C3,91,C9,3A, 2449
68 DATA 3C,AB,FE,00,C4,CB,91,C9,3E,07,CD,4 [4001]
5,91,21,AD,9C,C3,9A,91,CD, 2779
69 DATA C3,91,32,3C,AB,C9,11,06,00,CB,3F,C [2693]
B,3F,CB,3F,3C,47,19,10,FD, 2068
70 DATA C9,3E,C4,BD,DA,9A,91,C3,9E,91,3A,3 [2180]
6,AB,FE,00,28,05,01,82,7F, 2503
71 DATA ED,49,7E,C9,AF,32,39,AB,32,38,AB,3 [3240]
2,3C,AB,2A,51,AB,CD,6A,93, 2400
72 DATA FE,DD,CC,4B,93,FE,FD,CC,4B,93,CD,6 [3260]
A,93,FE,CB,CA,81,96,FE,ED, 3721
73 DATA CA,D1,96,21,FD,A0,06,0D,CD,15,93,2 [2129]
0,0D,3E,0D,90,07,07,26,A0, 1875
74 DATA C6,C9,6F,C3,9E,91,FE,40,D2,D2,94,F [3435]
E,02,21,00,9C,CA,AD,91,21, 2892
75 DATA 09,9C,FE,08,CA,AD,91,21,12,9C,FE,0 [3360]
A,CA,AD,91,FE,10,20,09,21, 2282
76 DATA 7B,A1,CD,A2,91,C3,35,92,21,1B,9C,F [4076]
E,12,CA,AD,91,FE,18,20,09, 2517
77 DATA 21,77,A1,CD,9A,91,C3,35,92,FE,1A,2 [4149]
1,24,9C,CA,AD,91,FE,22,21, 2557
78 DATA 05,A2,CA,42,97,FE,2A,CA,66,97,FE,3 [3832]
2,20,0C,CD,97,91,CD,19,92, 2562
79 DATA 3E,41,CD,45,91,C9,FE,3A,20,0B,CD,9 [2122]
7,91,3E,41,CD,45,91,C3,22, 2378
80 DATA 92,FE,10,38,04,D6,10,18,F8,FE,01,2 [3311]
0,09,CD,97,91,CD,ED,92,C3, 2558
81 DATA F9,91,FE,03,20,09,21,8B,A1,CD,9E,9 [3145]
1,C3,ED,92,FE,09,20,11,21, 2456
82 DATA AF,A1,CD,9E,91,CD,C9,92,3E,2C,CD,4 [2543]
5,91,C3,ED,92,FE,0B,20,05, 2801
83 DATA 21,8F,A1,18,DC,06,02,FE,04,20,0C,2 [3723]
1,8B,A1,CD,9E,91,32,38,AB, 2009
84 DATA C3,60,91,FE,05,20,05,21,8F,A1,18,E [4568]
E,FE,06,20,0C,CD,97,91,32, 2186
85 DATA 38,AB,CD,60,91,C3,E8,91,FE,00,20,3 [4146]
3,21,77,A1,CD,9A,91,CD,CB, 2807
86 DATA 91,CD,6A,93,E6,18,0F,0F,0F,47,3E,0 [2919]
4,90,47,21,15,A2,11,02,00, 1489
87 DATA 19,10,FD,CB,47,28,05,CD,8F,91,18,0 [3206]
3,CD,93,91,3E,2C,CD,45,91, 2155
88 DATA C3,35,92,D6,08,10,9C,C3,40,93,FE,8 [3095]
0,30,48,32,38,AB,FE,78,30, 2395
89 DATA 2C,FE,70,38,28,CD,97,91,32,38,AB,C [2368]
D,60,91,CD,37,93,CD,37,93, 2544
90 DATA 3E,2C,CD,45,91,32,39,AB,3A,3C,AB,F [2275]
E,00,CA,56,91,CD,6A,93,CD, 2442
91 DATA C3,91,C3,5C,91,CD,97,91,32,38,AB,C [2628]
D,60,91,3E,2C,CD,45,91,AF, 2696
92 DATA 32,39,AB,C3,56,91,FE,A0,30,13,D6,8 [3746]
0,21,2B,9C,CD,52,93,06,06, 2205
93 DATA CD,AF,91,32,38,AB,C3,56,91,FE,C0,3 [2694]
0,11,D6,A0,21,BF,A1,CD,5B, 2794
94 DATA 92,CD,61,93,32,38,AB,C3,56,91,FE,C [3032]
3,20,0C,21,87,A1,CD,9A,91, 2624
95 DATA CD,A7,8F,C3,FE,91,FE,D3,20,1B,21,A [3093]
3,A1,CD,9E,91,21,2D,9C,CD, 2937
96 DATA 93,91,CD,CB,91,CD,6A,93,CD,D3,91,2 [3375]
1,06,9C,C3,9A,91,FE,DB,20, 3058
97 DATA 19,21,49,9C,06,07,CD,AF,91,2A,51,A [3603]
B,CD,6A,93,CD,D3,91,3E,29, 2241
98 DATA CD,45,91,C3,C3,91,FE,CD,20,0C,21,7 [2886]
F,A1,CD,A2,91,CD,A7,8F,C3, 3000
99 DATA FE,91,06,0A,0E,02,21,50,9C,FE,E3,C [3852]
A,1C,93,21,83,A1,FE,C9,CA, 2540
100 DATA 9A,91,06,07,0C,21,5A,9C,FE,E9,CA, [3741]
1C,93,04,21,61,9C,0D,FE,F9, 2273
101 DATA CA,1C,93,21,69,9C,FE,EB,CA,AF,91, [3643]
D6,C0,FE,10,38,04,D6,10,18, 2672
102 DATA F8,FE,01,20,13,21,97,A1,CD,9E,91, [2999]
CD,F3,92,3E,03,BD,20,02,2D, 2334
103 DATA 2D,C3,93,91,FE,05,20,08,21,93,A1, [4218]
CD,A2,91,18,E7,06,02,FE,00, 2201
104 DATA 20,09,21,83,A1,CD,9E,91,C3,66,92, [3237]
FE,02,20,0F,21,87,A1,CD,9A, 2308
105 DATA 91,CD,A7,8F,CD,66,92,C3,F9,91,FE, [4049]
04,20,08,21,7F,A1,CD,A2,91, 2833
106 DATA 18,EB,FE,07,20,15,21,9B,A1,CD,9E, [3331]
91,CD,CB,91,CD,6A,93,E6,38, 2727
107 DATA 4F,CD,C3,91,C3,DE,91,D6,08,10,B7, [3960]
2A,51,AB,CD,6A,93,D6,C6,FE, 3025
108 DATA 19,30,0E,21,2B,9C,CD,52,93,06,05, [3829]
CD,AF,91,C3,E8,91,21,BF,A1, 2246
109 DATA D6,20,CD,5B,92,CD,61,93,2A,51,AB, [3523]
CD,6A,93,4F,CD,C3,91,C3,DE, 2930
110 DATA 91,CD,C3,91,CD,2E,93,CD,6A,93,FE, [3148]
40,30,19,FE,37,38,04,D6,08, 2528
111 DATA 18,05,FE,2F,D2,40,93,21,CF,A1,CD, [3510]
5B,92,CD,9E,91,C3,56,91,21, 2561
112 DATA E7,A1,CB,3F,CB,3F,CB,3F,CD,5B,92, [3179]
CD,9E,91,CD,CB,91,CD,6A,93, 3151
113 DATA E6,38,0F,0F,0F,C6,30,CD,45,91,3E, [3749]
2C,CD,45,91,CD,C3,91,C3,56, 2347
114 DATA 91,CD,C3,91,CD,6A,93,21,5E,A1,06, [3526]
15,CD,15,93,20,0D,3E,15,90, 2108
115 DATA 07,07,26,A1,C6,0A,6F,C3,9E,91,21, [2924]
9B,9C,06,07,CD,15,93,20,13, 1811
116 DATA 3E,07,90,47,04,21,6B,9C,11,06,00, [3297]
19,10,FD,06,06,C3,AF,91,FE, 1682
117 DATA 40,DA,40,93,FE,71,CA,40,93,FE,70, [2628]
CA,40,93,FE,80,D2,40,93,D6, 3069
118 DATA 40,FE,10,38,04,D6,10,18,F8,FE,02, [3663]
20,0B,21,A2,9C,06,07,CD,AF, 1939
119 DATA 91,C3,ED,92,FE,03,2A,51,AB,20,11, [2838]
CD,F6,92,E5,CD,97,91,CD,19, 2880
120 DATA 92,CD,CB,91,E1,C3,8F,92,FE,0A,20, [3730]
0B,21,B8,9C,06,07,CD,AF,91, 2626
121 DATA C3,ED,92,FE,0B,20,11,CD,F6,92,E5, [2481]
CD,97,91,E1,CD,8F,92,CD,CB, 3346
122 DATA 91,C3,22,92,06,02,FE,00,20,12,21, [2790]
9F,A1,CD,9A,91,CD,60,91,CD, 2340
123 DATA CB,91,21,A9,9C,C3,9E,91,FE,01,20, [2040]
0B,21,B0,9C,06,08,CD,AF,91, 2406
124 DATA C3,60,91,D6,08,10,D7,C3,40,93,3A, [3011]
36,AB,EE,01,32,36,AB,C3,95, 2436
125 DATA A3,21,6D,AB,7E,FE,20,CA,5C,98,E5, [4548]
CD,E6,9D,ED,43,44,AB,2A,3D, 2801
```

```
126 DATA AB,7E,FE,20,28,13,FE,2D,C2,61,A4, [3461]
23,7E,01,FF,FF,FE,20,28,05, 2399
127 DATA E5,CD,E6,9D,E1,ED,43,46,AB,06,08, [2692]
2A,28,AB,AF,BC,20,03,21,02, 2291
128 DATA 29,22,85,B2,C5,2A,44,AB,7C,CD,7A, [3454]
9D,7D,CD,7A,9D,11,86,B2,06, 2416
129 DATA 08,2A,44,AB,1A,3C,12,CD,71,9B,10, [4987]
F8,CD,4A,9B,1B,1A,FE,16,20, 1925
130 DATA 1B,04,78,21,02,29,11,16,4F,CD,50, [3071]
BC,3E,29,32,86,B2,3E,20,06, 1383
131 DATA 27,CD,5A,BB,10,FB,18,07,3C,32,28, [3111]
AB,32,85,B2,3E,29,32,86,B2, 1966
132 DATA 32,29,AB,00,C1,21,45,AB,3A,47,AB, [4693]
BE,38,15,2B,3A,46,AB,BE,38, 1883
133 DATA 0E,10,9D,CD,18,BB,06,08,CD,E0,9D, [4496]
FE,53,20,91,00,CD,1A,9B,26, 2141
134 DATA 2E,CA,61,A4,2D,22,85,B2,CD,8A,BB, [5318]
CD,18,BB,F5,CD,8D,BB,F1,2A, 2906
135 DATA 85,B2,FE,F0,CA,38,99,FE,F1,CA,41, [2870]
99,FE,F2,CA,4F,99,FE,F3,CA, 3760
136 DATA 64,99,CD,BA,9B,FE,53,CA,95,A3,FE, [4379]
30,38,D2,FE,47,30,CE,FE,3A, 3109
137 DATA 38,0B,FE,41,38,C6,CD,20,9B,D6,07, [3748]
18,03,CD,20,9B,D6,30,32,30, 2032
138 DATA AB,E5,CD,DB,9B,18,0F,00,00,00,00, [3096]
00,00,00,00,00,00,00,00,00, 1018
139 DATA 00,00,E1,22,85,B2,7C,D6,2E,06,03, [3711]
0E,00,B8,38,06,0C,04,04,04, 1247
140 DATA 18,F7,06,00,2A,44,AB,09,E5,21,86, [2958]
B2,34,CD,60,BB,FE,20,20,16, 2021
141 DATA 00,35,35,CD,60,BB,34,34,34,CD,03, [3634]
9B,07,07,07,07,2A,30,AB,B5, 1583
142 DATA 18,0C,CD,03,9B,67,3A,30,AB,07,07, [3465]
07,07,B4,E1,00,00,77,32,30, 1429
143 DATA AB,3A,31,AB,FE,00,CA,68,98,3E,48, [3118]
81,2A,85,B2,32,86,B2,3A,30, 2245
144 DATA AB,FE,20,30,02,3E,2E,CD,5A,BB,22, [3565]
85,B2,C3,68,98,3E,02,BD,CA, 2348
145 DATA 68,98,2D,18,08,3A,28,AB,2C,BD,CA, [3770]
68,98,22,85,B2,C3,68,98,3E, 2151
146 DATA 2E,BC,CA,68,98,25,22,85,B2,CD,60, [3662]
BB,FE,20,C2,68,98,25,18,E5, 2588
147 DATA 3E,46,BC,CA,68,98,24,22,85,B2,CD, [4328]
60,BB,FE,20,C2,68,98,24,C3, 2614
148 DATA 49,99,E1,22,3F,AB,21,6D,AB,CD,79, [4134]
9B,ED,43,44,AB,CD,79,9B,C3, 2732
149 DATA C3,8E,AF,B8,C2,61,A4,2A,44,AB,ED, [4112]
5B,46,AB,7A,BC,DA,61,A4,20, 2822
150 DATA 05,7B,BD,DA,61,A4,71,23,7C,BA,38, [3025]
FA,7B,BD,30,F6,C3,95,A3,21, 2706
151 DATA 02,00,22,85,B2,0E,16,3E,20,06,28, [3489]
CD,5A,BB,10,FB,2C,22,85,B2, 1661
152 DATA 0D,41,10,F1,21,02,00,22,56,AB,AF, [2812]
32,58,AB,22,5A,AB,C3,95,A3, 1947
153 DATA 21,6D,AB,CD,87,9B,C3,AB,9B,00,21, [3746]
ED,99,71,23,70,CD,D0,07,32, 2482
154 DATA 10,AB,22,16,AB,22,14,AB,21,11,AB, [3372]
ED,73,5B,AB,31,F0,BF,00,CD, 2159
155 DATA 8B,9B,23,23,CD,94,9B,DD,22,44,AB, [3279]
ED,4B,44,AB,FD,22,44,AB,ED, 2680
156 DATA 5B,44,AB,CD,8B,9B,ED,4B,5B,AB,CD, [3592]
8B,9B,D9,08,32,1F,AB,22,23, 2448
157 DATA AB,22,25,AB,00,21,20,AB,CD,8B,9B, [4487]
23,23,CD,94,9B,F1,08,C1,D1, 2377
158 DATA E1,D9,C3,69,A2,00,21,6D,AB,CD,79, [2328]
9B,ED,43,6B,AB,CD,79,9B,ED, 2998
159 DATA 43,6D,AB,CD,87,9B,ED,43,7E,AB,2A, [3640]
6D,AB,ED,5B,6B,AB,B7,ED,52, 2873
160 DATA 23,44,4D,2A,7E,AB,ED,52,22,78,AB, [3828]
38,11,ED,42,30,0D,2A,7E,AB, 1939
161 DATA 09,2B,EB,2A,6D,AB,ED,B8,18,07,EB, [2714]
ED,5B,7E,AB,ED,B0,18,05,00, 2363
162 DATA 64,00,C8,00,3A,8F,9A,FE,00,CA,95, [4025]
A3,2A,90,9A,22,44,AB,CD,AD, 2414
163 DATA 8F,AF,32,3A,AB,CD,78,93,3A,3A,AB, [2794]
FE,00,C4,CE,9A,2A,51,AB,ED, 2697
164 DATA 5B,92,9A,B7,ED,52,38,E5,00,00,00, [3885]
00,AF,32,37,AB,32,3A,AB,C3, 2103
165 DATA 95,A3,2A,51,AB,2B,56,2B,5E,2A,6B, [4167]
AB,2B,ED,52,D0,2A,6D,AB,B7, 2267
166 DATA ED,52,D8,2A,78,AB,19,EB,2A,51,AB, [4022]
2B,72,2B,73,C9,00,00,E2,FE, 2418
167 DATA 8E,C5,F5,ED,4B,56,AB,7C,C3,E2,8E, [2573]
00,CD,60,BB,D6,30,FE,0A,D8, 3070
168 DATA D6,07,C9,CD,00,9B,07,07,07,07,57, [2516]
34,CD,00,9B,B2,34,C9,2A,28, 1822
169 DATA AB,AF,BC,C9,47,3E,46,BC,78,C2,5A, [4882]
BB,E1,C3,68,98,3A,31,AB,EE, 2909
170 DATA 01,32,31,AB,FE,00,28,08,3E,4F,32, [3271]
18,98,C3,95,A3,3E,45,18,F6, 1848
171 DATA 3E,20,CD,45,91,C9,22,44,AB,3A,31, [3273]
AB,FE,00,C8,3E,48,32,86,B2, 2215
172 DATA 01,08,00,B7,ED,42,41,C5,CD,6A,93, [3572]
C1,FE,20,30,02,3E,2E,CD,5A, 2147
173 DATA BB,23,10,EF,C9,C5,CD,6A,93,C1,C3, [2965]
79,9D,CD,E6,9D,2A,3D,AB,7E, 2991
174 DATA FE,2C,C2,A3,8F,23,C9,CD,E6,9D,C9, [4308]
70,23,71,23,72,23,73,23,C9, 2622
175 DATA 3E,01,36,00,30,01,77,23,36,00,20, [3399]
01,77,23,C9,21,6D,AB,CD,79, 1401
176 DATA 9B,18,45,D9,E5,D5,C5,D9,08,F5,08, [3644]
31,D0,BF,C3,E6,99,00,CD,E0, 3037
177 DATA 9D,FE,0D,C0,D1,3A,28,AB,3D,BD,C2, [3512]
68,98,CD,DB,9B,2A,44,AB,11, 2671
178 DATA 08,00,19,22,44,AB,44,4D,C3,D9,97, [2268]
26,29,22,85,B2,21,86,B2,CD, 1988
179 DATA 0B,9B,32,45,AB,CD,0B,9B,32,44,AB, [3059]
C9,ED,43,90,9A,CD,87,9B,ED, 2651
180 DATA 43,92,9A,7C,C3,D3,91,00,4C,44,2E, [3124]
28,42,43,29,2C,41,45,58,2E, 1758
181 DATA 41,46,2C,41,46,27,4C,44,2E,41,2C, [3324]
28,42,43,29,4C,44,2E,28,44, 1164
182 DATA 45,29,2C,41,4C,44,2E,41,2C,28,44, [3300]
45,29,28,26,2C,26,41,44,44, 1097
183 DATA 2E,41,2C,41,44,43,2E,41,2C,53,55, [2791]
42,2E,41,2C,53,42,43,2E,41, 1226
184 DATA 2C,49,4E,2E,41,2C,28,26,45,58,2E, [3158]
28,53,50,29,2C,48,4C,4A,50, 1221
185 DATA 2E,28,48,4C,29,4C,44,2E,53,50,2C, [3951]
48,4C,45,58,2E,44,45,2C,48, 1276
186 DATA 4C,49,4D,2E,30,2E,2E,4C,44,2E,49, [2834]
2C,41,4C,44,2E,41,2C,49,49, 1229
187 DATA 4D,2E,31,2E,2E,49,4D,2E,32,2E,2E, [3202]
4C,44,2E,52,2C,41,4C,44,2E, 1173
188 DATA 41,2C,52,46,47,57,56,5E,4F,5F,53, [4407]
42,43,2E,48,4C,2C,2C,28,43, 1378
189 DATA 29,3F,3F,3F,4F,55,54,2E,28,43,29, [2995]
2C,41,44,43,2E,48,4C,2C,00, 1154
190 DATA FE,04,C2,E0,9E,78,FE,04,D2,E0,9E, [2452]
0F,0F,0F,0F,B4,0E,ED,47,18, 2390
191 DATA 6D,2A,5E,AB,77,23,C3,4F,9D,FE,01, [3377]
20,03,7C,18,44,FE,03,20,09, 1805
192 DATA AF,B8,C2,E0,9E,41,4D,18,51,FE,0D, [4821]
C2,E0,9E,AF,B8,C2,E0,9E,41, 3025
193 DATA 4A,18,48,16,CB,FE,01,20,06,7C,B0, [3327]
47,4A,18,37,FE,0D,C2,E0,9E, 2055
194 DATA AF,B8,C2,E0,9E,43,18,1E,CB,08,CB, [4883]
08,CB,08,CB,08,18,06,CB,20, 2165
195 DATA CB,20,CB,20,B0,ED,4B,5E,AB,02,03, [4106]
ED,43,5E,AB,C3,89,A4,3A,41, 2416

196 DATA AB,2A,5E,AB,77,23,72,23,18,0D,2A, [2726]
5E,AB,18,08,2A,5E,AB,3A,41, 1587
197 DATA AB,77,23,71,23,70,23,22,5E,AB,44, [3570]
4D,C3,89,A4,FE,01,C2,DF,9E, 2390
198 DATA 78,FE,01,C2,DF,9E,06,03,C9,0A,FE, [3770]
00,C8,CD,5A,BB,03,18,F6,13, 2398
199 DATA ED,53,85,B2,7E,23,4F,0F,0F,0F,0F, [3595]
E6,0F,CD,8B,9D,79,E6,0F,CD, 2248
200 DATA 8B,9D,C9,FE,0A,38,02,C6,07,C6,30, [3389]
CD,45,91,C9,21,C8,B8,11,C8, 2524
201 DATA 00,06,08,C5,01,EB,07,09,01,15,00, [3683]
ED,B0,C1,10,F3,3E,01,47,21, 1517
202 DATA 02,00,11,16,27,CD,50,BC,21,C8,00, [2936]
11,F4,BE,06,08,C5,01,EB,07, 1691
203 DATA EB,09,EB,01,15,00,ED,B0,C1,10,F1, [2937]
11,16,14,ED,53,85,B2,3E,20, 2148
204 DATA 06,14,CD,5A,BB,10,FB,C9,FE,60,D8, [3700]
D6,20,C9,E5,7E,FE,2F,38,03, 2704
205 DATA 23,18,F8,FE,26,28,F9,54,5D,22,3D, [3686]
AB,E1,7E,FE,26,20,16,23,7B, 2186
206 DATA 95,FE,05,D4,7F,9E,AF,32,50,AB,DD, [4276]
21,00,00,FD,21,48,AB,18,1E, 2218
207 DATA FE,30,DC,7F,9E,FE,3A,D4,7F,9E,3E, [2826]
01,32,50,AB,7B,95,FE,06,D4, 2724
208 DATA 7F,9E,DD,21,00,00,FD,21,61,AB,1B, [3258]
1A,FE,30,28,26,3A,50,AB,FE, 2089
209 DATA 01,28,0C,1A,CD,71,9E,FE,3A,38,08, [3908]
D6,07,18,04,1A,CD,78,9E,D6, 1903
210 DATA 30,D5,FD,56,01,FD,5E,00,47,DD,19, [3618]
10,FC,D1,FD,23,FD,23,7B,BD, 2630
211 DATA 20,CC,DD,22,51,AB,ED,4B,51,AB,3E, [4152]
0B,C9,FE,47,30,0A,FE,40,D0, 2490
212 DATA FE,3A,30,03,FE,2F,D0,3A,42,AB,D1, [4031]
D1,FE,01,C2,60,A4,AF,32,42, 2585
213 DATA AB;C3,61,A4,FE,01,C0,78,FE,07,3E, [4399]
00,C9,11,14,AB,1A,BE,C0,13, 2353
214 DATA 23,10,F9,C9,C5,E5,41,CD,99,9E,E1, [4727]
```

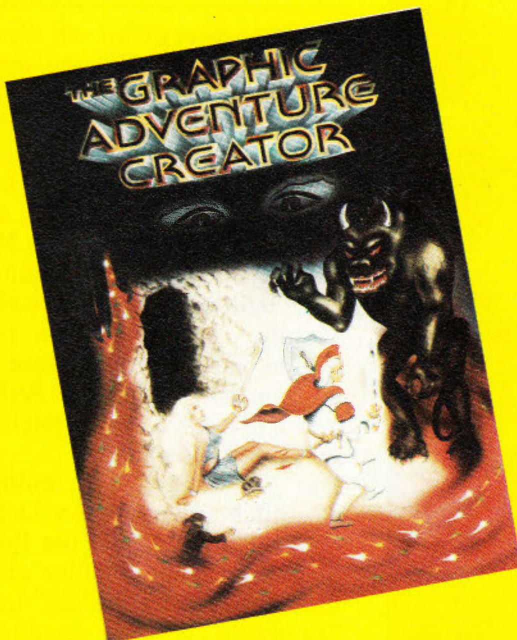
```
C1,C8,59,16,00,19,10,F1,AF, 2694
215 DATA B9,C9,FE,04,C0,78,FE,02,3E,00,C9, [3354]
21,14,AB,3E,2E,06,08,77,23, 1975
216 DATA 10,FC,2A,5C,AB,11,14,AB,06,09,7E, [3721]
FE,20,C8,FE,2C,C8,12,23,13, 1978
217 DATA 10,F4,D1,D1,3E,07,CD,5A,BB,ED,4B, [4443]
5E,AB,C3,89,A4,CD,BF,9E,3E, 2918
218 DATA 09,90,32,43,AB,FE,00,C8,23,22,5C, [2830]
AB,FE,01,28,26,FE,02,28,53, 1939
219 DATA FE,03,CA,87,9F,FE,04,CA,9F,9F,FE, [3865]
05,20,08,CD,F4,9F,CD,0C,A0, 2815
220 DATA 18,34,CD,C4,9F,3A,43,AB,CD,F4,9F, [4210]
C3,DF,9E,21,F7,A1,06,08,4F, 2650
221 DATA CD,A4,9E,20,04,05,3E,01,C9,06,02, [2971]
CD,A4,9E,20,03,3E,06,C9,21, 1704
222 DATA 0F,A2,06,08,0C,CD,A4,9E,20,04,05, [4387]
3E,02,C9,21,14,AB,CD,E6,9D, 1852
223 DATA 3E,03,C9,21,01,A2,06,05,0E,02,CD, [2841]
A4,9E,20,04,05,3E,04,C9,06, 1330
224 DATA 02,CD,A4,9E,20,0E,3E,02,B8,28,04, [3424]
0E,DD,18,02,0E,FD,3E,05,C9, 1663
225 DATA 06,08,CD,A4,9E,20,CB,05,3E,02,C9, [3878]
CD,0C,A0,06,02,0E,03,CD,A4, 1817
226 DATA 9E,20,04,78,C6,08,C9,3A,43,AB,CD, [3044]
F4,9F,18,AF,21,29,A2,06,06, 2072
227 DATA 0E,04,CD,A4,9E,20,0E,05,78,FE,02, [3077]
20,05,3E,01,06,06,C9,3E,0C, 1359
228 DATA C9,CD,0C,A0,3A,43,AB,CD,F4,9F,18, [4150]
8A,21,41,A2,06,02,0E,04,CD, 2135
229 DATA A4,9E,C0,CB,40,28,04,3E,DD,18,02, [3480]
3E,FD,32,41,AB,3A,43,AB,21, 2064
230 DATA 13,AB,85,6F,7E,FE,29,E1,C2,DF,9E, [3772]
21,18,AB,CD,E6,9D,3E,0D,C9, 2751
231 DATA 3D,21,14,AB,5F,16,00,7E,FE,28,C0, [3783]
19,7E,FE,29,C0,E1,21,15,AB, 2102
232 DATA CD,E6,9D,C9,21,1F,A2,06,02,48,CD, [3179]
A4,9E,C0,78,C6,06,32,53,AB, 2446
233 DATA E1,21,16,AB,CD,E6,9D,AF,B8,DA,DF, [3886]
9E,3A,53,AB,57,FE,07,28,0A, 2711
234 DATA 3E,81,B9,DA,DF,9E,0D,0D,7A,C9,3E, [2686]
7E,B9,DA,DF,9E,AF,0C,0C,91, 2640
235 DATA 4F,7A,3C,C9,2A,2A,2A,00,43,50,43, [2938]
2D,4D,6F,6E,2E,00,76,6F,6E, 1530
236 DATA 00,50,2E,42,65,6E,65,73,00,41,2E, [3141]
42,43,2E,44,45,2E,48,4C,2E, 1286
237 DATA 43,2E,5A,2E,53,50,00,5B,41,5D,3D, [2364]
41,73,73,65,6D,62,6C,65,72, 1648
238 DATA 2E,5B,44,5D,3D,44,69,73,61,73,73, [3753]
2E,5B,4D,5D,3D,4D,6F,6E,2E, 1686
239 DATA 5B,52,5D,3D,52,75,6E,2E,5B,56,5D, [3590]
3D,56,65,72,73,63,68,69,65, 1838
240 DATA 62,65,6E,2E,5B,46,5D,3D,46,75,65, [4302]
6C,6C,65,6E,2E,5B,42,5D,3D, 1742
241 DATA 42,61,73,69,63,00,46,65,68,6C,65, [2666]
72,00,43,43,46,2E,43,50,4C, 1553
242 DATA 2E,44,41,41,2E,44,49,2E,2E,45,49, [2813]
2E,2E,45,58,58,2E,48,41,4C, 1261
243 DATA 54,4E,4F,50,2E,52,4C,41,2E,52,4C, [2636]
43,41,52,52,41,2E,52,52,43, 1432
244 DATA 41,53,43,46,2E,3F,2F,27,F3,FB,D9, [2344]
76,00,17,07,1F,0F,37,43,50, 1587
245 DATA 44,2E,43,50,44,52,43,50,49,2E,43, [4024]
50,49,52,49,4E,44,2E,49,4E, 1395
246 DATA 44,52,49,4E,49,2E,49,4E,49,52,4C, [3668]
44,44,2E,4C,44,44,52,4C,44, 1422
247 DATA 49,2E,4C,44,49,52,4E,45,47,2E,4F, [4678]
54,44,52,4F,54,49,52,4F,55, 1477
248 DATA 54,44,4F,55,54,49,52,45,54,49,52, [3697]
45,54,4E,52,4C,44,2E,52,52, 1530
249 DATA 44,2E,A9,B9,A1,B1,AA,BA,A2,B2,A8, [5027]
B8,A0,B0,44,BB,B3,AB,A3,4D, 3035
250 DATA 45,6F,67,4C,44,2E,2E,4A,52,2E,2E, [4358]
44,4A,4E,5A,43,41,4C,4C,52, 1443
251 DATA 45,54,2E,4A,50,2E,2E,49,4E,43,2E, [3391]
44,45,43,2E,50,55,53,48,50, 1359
252 DATA 4F,50,2E,52,53,54,2E,49,4E,2E,2E, [2861]
4F,55,54,2E,49,4D,2E,2E,45, 1348
253 DATA 58,2E,2E,41,44,44,2E,41,44,43,2E, [2245]
53,55,42,2E,53,42,43,2E,41, 1280
254 DATA 4E,44,2E,58,4F,52,2E,4F,52,2E,2E, [4378]
43,50,2E,2E,52,4C,43,2E,52, 1332
255 DATA 52,43,2E,52,4C,2E,2E,52,52,2E,2E, [2542]
53,4C,41,2E,53,52,41,2E,53, 1330
256 DATA 52,4C,2E,42,49,54,2E,52,45,53,2E, [2628]
53,45,54,2E,41,A3,4C,48,45, 1480
257 DATA 44,43,42,49,52,41,46,53,50,48,4C, [3992]
44,45,42,43,49,59,49,58,4D, 1472
258 DATA 2E,50,2E,50,45,50,4F,43,2E,4E,43, [2612]
5A,2E,4E,5A,24,2B,24,2D,41, 1267
259 DATA 46,27,28,43,29,28,49,59,29,28,49, [2901]
58,29,28,53,50,29,28,48,4C, 1174
260 DATA 29,28,44,45,29,28,42,43,29,28,49, [2485]
59,2B,28,49,58,2B,FE,00,C8, 1416
261 DATA FE,FF,C3,CF,90,41,2E,42,43,2E,44, [3624]
45,2E,48,4C,2E,43,2E,5A,2E, 1971
262 DATA 49,58,2E,49,59,2E,53,50,2E,AF,21, [4188]
28,AB,06,58,77,23,10,FC,21, 1592
263 DATA 01,00,22,61,AB,22,48,AB,2E,0A,22, [2663]
63,AB,2E,64,22,65,AB,21,E8, 1657
264 DATA 03,22,67,AB,21,10,27,22,69,AB,26, [3565]
00,2E,10,22,4A,AB,21,00,01, 1122
265 DATA 22,4C,AB,26,10,22,4E,AB,3E,02,32, [2582]
56,AB,32,5A,AB,3E,02,CD,0E, 1583
266 DATA BC,21,00,1D,22,85,B2,01,48,A0,CD, [2250]
69,9D,21,86,B2,34,03,CD,69, 2005
267 DATA 9D,34,01,48,A0,CD,69,9D,36,45,01, [3155]
55,A0,CD,69,9D,34,CD,9C,BB, 2345
268 DATA 01,59,A0,CD,69,9D,CD,9C,BB,21,04, [1058]
48,06,09,22,85,B2,11,51,A2, 1994
269 DATA 1A,FE,2E,28,08,E5,CD,5A,BB,E1,13, [4795]
18,F3,23,22,85,B2,13,10,EC, 2247
270 DATA E5,CD,9C,BB,E1,3A,59,AB,FE,00,20, [3215]
0B,3E,01,32,59,AB,00,23,06, 2031
271 DATA 06,18,CF,AF,32,59,AB,21,04,4A,22, [3170]
85,B2,06,10,3E,3D,CD,5A,BB, 1805
272 DATA 3E,26,CD,5A,BB,23,22,85,B2,10,F0, [3731]
21,0D,4A,22,85,B2,3E,20,CD, 1982
273 DATA 5A,BB,CD,5A,BB,21,10,AB,11,04,4C, [4226]
ED,53,85,B2,CD,78,9D,06,03, 2198
274 DATA CD,73,9D,CD,78,9D,10,F8,06,02,CD, [4243]
73,9D,10,FB,06,03,CD,73,9D, 2461
275 DATA CD,78,9D,10,F8,13,CD,73,9D,06,03, [3795]
CD,73,9D,CD,78,9D,10,F8,CD, 2679
276 DATA 73,9D,CD,73,9D,CD,9C,BB,01,73,A0, [4009]
11,18,00,ED,53,85,B2,CD,69, 2555
277 DATA 9D,CD,9C,BB,3E,20,32,1C,AB,3E,20, [3558]
21,6B,AB,06,15,77,23,10,FC, 1902
278 DATA 21,6B,AB,22,5C,AB,ED,5B,56,AB,ED, [4425]
53,85,B2,CD,8A,BB,CD,18,BB, 2770
279 DATA F5,CD,8D,BB,F1,FE,0D,28,58,FE,7F, [4116]
CA,82,90,FE,10,CA,AA,90,FE, 3311
280 DATA F2,28,20,FE,F3,28,31,FE,1F,38,DB, [4533]
CD,5A,BB,2A,5C,AB,77,23,22, 2435
281 DATA 5C,AB,3A,5B,AB,3C,FE,14,28,2F,32, [2601]
5B,AB,18,C3,3A,5B,AB,FE,00, 2109
282 DATA 28,BC,21,86,B2,35,3D,32,5B,AB,21, [3491]
5C,AB,35,18,AE,3A,5B,AB,FE, 2120
283 DATA 14,28,A7,21,86,B2,34,3C,32,5B,AB, [3556]
21,5C,AB,34,18,99,CD,1B,A4, 1917
284 DATA C3,FD,AA,3A,5A,AB,3C,FE,17,38,14, [3932]
3A,58,AB,FE,00,28,25,CD,97, 2354
285 DATA 9D,3E,14,32,57,AB,AF,32,5B,AB,C9, [3637]
32,5A,AB,5F,16,00,3A,58,AB, 1980
286 DATA FE,00,28,02,16,14,AF,32,5B,AB,ED, [3272]
53,56,AB,C9,3C,32,58,AB,21, 2005
287 DATA 02,14,22,56,AB,26,00,22,5A,AB,EB, [3470]
C9,D1,3E,07,CD,5A,BB,01,C2, 2037
288 DATA A0,ED,5B,56,AB,ED,53,85,B2,CD,69, [2940]
9D,CD,1B,A4,C3,95,A3,21,6D, 2888
289 DATA AB,3E,01,32,42,AB,CD,E6,9D,AF,32, [3302]
42,AB,ED,5B,56,AB,ED,43,5E, 2558
290 DATA AB,ED,53,85,B2,78,41,CD,7A,9D,78, [2125]
CD,7A,9D,3E,01,32,60,AB,2A, 2497
291 DATA 85,B2,24,22,56,AB,3E,05,32,5B,AB, [3293]
C3,95,A3,21,6B,AB,7E,FE,57, 2302
292 DATA CA,03,90,FE,5A,CA,21,90,FE,53,20, [2908]
0E,23,7E,2B,FE,20,20,07,AF, 2159
293 DATA 32,60,AB,C3,95,A3,E5,22,5C,AB,CD, [3030]
BF,9E,23,22,5C,AB,E1,21,73, 2609
294 DATA A1,06,21,0E,04,CD,A4,9E,C2,7B,A5, [3031]
78,FE,21,CA,85,A9,FE,20,CA, 2626
295 DATA 32,A9,FE,1F,CA,14,A7,FE,1E,CA,5D, [4046]
A8,FE,1D,CA,BA,A6,FE,1C,CA, 2961
296 DATA DE,A8,FE,1B,CA,8B,A6,FE,1A,CA,80, [3825]
A6,FE,19,CA,39,A6,FE,18,CA, 3138
297 DATA 30,A6,FE,17,CA,D4,A6,FE,16,CA,F5, [4339]
A5,FE,15,CA,D2,A5,FE,14,CA, 3287
298 DATA 59,A6,FE,13,CA,85,A8,FE,12,CA,CE, [3528]
A7,FE,11,CA,37,A8,FE,10,CA, 3046
299 DATA BA,A7,FE,0F,CA,E2,A6,FE,0E,CA,A4, [3823]
A7,FE,0D,CA,09,A7,FE,0C,CA, 3130
300 DATA D2,A8,FE,0B,CA,AF,A7,FE,03,CA,7E, [3056]
A7,FE,02,CA,70,A7,FE,01,CA, 3133
301 DATA 77,A7,D6,03,47,21,20,A7,11,0A,00, [2931]
19,10,FD,E9,21,0A,A1,06,15, 1591
302 DATA 0E,04,CD,A4,9E,20,1B,21,5E,A1,3E, [3744]
15,90,06,00,4F,09,7E,2A,5E, 1475
303 DATA AB,36,ED,23,77,23,22,5E,AB,44,4D, [3856]
```

```
C3,89,A4,21,C9,A0,06,0D,0E, 2018
304 DATA 04,CD,A4,9E,28,0C,3E,07,CD,5A,BB, [3007]
ED,4B,5E,AB,C3,89,A4,21,FD, 2493
305 DATA A0,3E,0D,90,06,00,4F,09,7E,2A,5E, [2913]
AB,77,23,22,5E,AB,44,4D,C3, 1699
306 DATA 89,A4,CD,EC,9E,32,55,AB,ED,43,3F, [4402]
AB,CD,EC,9E,FE,01,C2,E0,9E, 3174
307 DATA 78,32,41,AB,ED,4B,3F,AB,3A,55,AB, [3871]
2E,D3,26,41,18,13,CD,EC,9E, 2268
308 DATA FE,01,C2,E0,9E,78,32,41,AB,CD,EC, [2537]
9E,26,40,2E,DB,FE,09,20,12, 2516
309 DATA 3A,41,AB,FE,06,CA,E0,9E,07,07,07, [3728]
B4,47,0E,ED,C3,42,9D,FE,0B, 2344
310 DATA C2,E0,9E,3A,41,AB,FE,07,C2,E0,9E, [3274]
3E,03,C3,DD,9C,CD,EC,9E,16, 2965
311 DATA C1,1E,E1,18,07,CD,EC,9E,16,C5,1E, [3794]
E5,FE,05,20,04,43,C3,42,9D, 2336
312 DATA FE,04,C2,E0,9E,3D,B8,CA,E0,9E,30, [3688]
01,05,7A,C3,18,9D,CD,EC,9E, 2814
313 DATA FE,03,C2,E0,9E,41,AF,0E,ED,B8,20, [3692]
05,06,46,C3,42,9D,3C,B8,20, 2315
314 DATA 05,06,56,C3,42,9D,3C,B8,C2,E0,9E, [3711]
06,5E,C3,42,9D,CD,EC,9E,21, 2485
315 DATA 0B,05,11,2B,35,18,09,CD,EC,9E,21, [4018]
03,04,11,23,34,FE,01,20,04, 1196
316 DATA 7C,C3,22,9D,FE,04,20,08,B8,CA,E0, [3390]
9E,7D,C3,18,9D,FE,0D,20,05, 2381
317 DATA 41,4A,C3,47,9D,FE,05,C2,E0,9E,43, [3582]
C3,42,9D,CD,EC,9E,FE,00,20, 2767
318 DATA 05,3E,C9,C3,29,9D,FE,02,C4,5B,9D, [3418]
78,07,07,07,F6,C0,C3,29,9D, 2333
319 DATA CD,EC,9E,FE,03,C2,E0,9E,3E,C7,41, [4018]
C3,28,9D,CD,EC,9E,FE,01,20, 3036
320 DATA 12,3E,07,B8,C2,E0,9E,CD,EC,9E,26, [2678]
98,2E,DE,16,9E,C3,DD,9C,CD, 2861
321 DATA B6,9E,C2,E0,9E,CD,EC,9E,26,42,C3, [4339]
C0,9C,CD,EC,9E,21,EE,A8,16, 3222
322 DATA AE,C3,DD,9C,CD,EC,9E,1E,10,FE,03, [2902]
CA,3B,A9,FE,08,00,C2,E0,9E, 2916
323 DATA 41,0E,10,C3,42,9D,CD,EC,9E,26,38, [3610]
1E,3E,C3,FF,9C,CD,EC,9E,26, 2541
324 DATA 28,1E,2E,C3,FF,9C,CD,EC,9E,26,20, [3686]
1E,26,C3,FF,9C,CD,EC,9E,26, 2702
325 DATA 18,1E,1E,C3,FF,9C,CD,EC,9E,26,10, [3545]
1E,16,C3,FF,9C,CD,EC,9E,26, 2638
326 DATA 08,1E,0E,C3,FF,9C,CD,EC,9E,26,00, [2755]
1E,06,C3,FF,9C,CD,EC,9E,16, 2558
327 DATA 80,18,0C,CD,EC,9E,16,C0,18,05,CD, [3315]
EC,9E,16,40,FE,03,C2,E0,9E, 2524
328 DATA 79,FE,08,D2,E0,9E,07,07,07,B2,32, [3240]
54,AB,CD,EC,9E,F5,3A,54,AB, 2636
329 DATA 67,F6,06,5F,F1,C3,FF,9C,CD,EC,9E, [2990]
21,E6,A0,16,A6,C3,DD,9C,CD, 3284
330 DATA EC,9E,21,FE,B8,16,BE,C3,DD,9C,CD, [2879]
EC,9E,CD,90,9E,C2,E0,9E,CD, 3536
331 DATA EC,9E,21,D6,90,16,96,C3,DD,9C,CD, [3237]
EC,9E,CD,90,9E,20,0B,CD,EC, 3119
332 DATA 9E,21,C6,80,16,86,C3,DD,9C,CD,B6, [3600]
9E,20,13,CD,EC,9E,FE,04,C2, 2892
333 DATA E0,9E,78,FE,04,CA,E0,9E,3E,09,C3, [4647]
18,9D,FE,05,C2,E0,9E,79,32, 2797
334 DATA 3F,AB,CD,EC,9E,FE,04,20,11,3E,02, [4504]
B8,CA,E0,9E,87,B8,CA,E0,9E, 2875
335 DATA 3A,3F,AB,4F,18,0E,FE,05,C2,E0,9E, [4185]
3A,3F,AB,B9,C2,E0,9E,06,02, 2305
336 DATA CB,08,CB,08,CB,08,CB,08,3E,09,B0, [2373]
47,C3,42,9D,CD,EC,9E,FE,01, 2434
337 DATA 20,11,3E,07,B8,C2,E0,9E,CD,EC,9E, [3268]
21,CE,88,16,8E,C3,DD,9C,CD, 2793
338 DATA B6,9E,C2,E0,9E,CD,EC,9E,26,4A,C3, [4849]
C0,9C,CD,EC,9E,FE,03,20,05, 3063
339 DATA 3E,CD,C3,D5,9C,FE,02,C4,5B,9D,78, [3957]
07,07,07,F6,C4,32,54,AB,CD, 2624
340 DATA EC,9E,FE,03,C2,E0,9E,3A,54,AB,C3, [3828]
D5,9C,CD,EC,9E,FE,04,20,25, 3030
341 DATA 78,FE,01,20,0E,CD,EC,9E,CD,B6,9E, [4664]
C2,E0,9E,3E,EB,C3,29,9D,FE, 3085
342 DATA 04,C2,E0,9E,CD,EC,9E,FE,0A,C2,E0, [4169]
9E,3E,08,C3,29,9D,FE,0C,C2, 2942
343 DATA E0,9E,78,FE,03,C2,E0,9E,CD,EC,9E, [2760]
57,CD,B6,9E,3E,E3,CA,29,9D, 3255
344 DATA 47,7A,FE,05,C2,E0,9E,C3,42,9D,CD, [2618]
EC,9E,26,B0,2E,F6,16,B6,C3, 2950
345 DATA DD,9C,CD,EC,9E,FE,02,20,14,78,07, [2491]
07,07,F6,C2,32,41,AB,CD,EC, 2592
346 DATA 9E,FE,03,C2,E0,9E,C3,47,9D,FE,03, [4009]
20,05,3E,C3,C3,D5,9C,FE,01, 2784
347 DATA 20,11,78,C6,02,FE,03,28,D9,FE,08, [2282]
C2,E0,9E,3E,E9,C3,29,9D,FE, 2663
348 DATA 0C,C2,E0,9E,78,06,E9,FE,05,20,05, [4195]
0E,FD,C3,42,9D,FE,04,C2,E0, 2604
349 DATA 9E,0E,DD,C3,42,9D,CD,EC,9E,FE,03, [3468]
1E,18,20,21,60,69,ED,4B,5E, 2393
350 DATA AB,ED,42,2B,2B,3E,80,BD,38,07,AF, [2860]
BC,C2,E0,9E,18,06,3E,FF,BC, 2476
351 DATA C2,E0,9E,4B,45,C3,42,9D,FE,08,20, [4614]
05,41,4B,C3,42,9D,FE,02,C4, 2447
352 DATA 5B,9D,78,07,07,07,F6,20,32,54,AB, [3401]
CD,EC,9E,ED,5B,54,AB,FE,08, 2416
353 DATA 28,E2,FE,03,28,B9,C3,E0,9E,CD,EC, [2716]
9E,ED,43,3F,AB,FE,04,20,53, 2835
354 DATA B8,CA,E0,9E,CD,EC,9E,ED,5B,3F,AB, [4984]
FE,03,20,0A,7A,0F,0F,0F,0F, 2410
355 DATA F6,01,C3,D5,9C,FE,0B,20,1A,3A,40, [3727]
AB,FE,02,20,05,3E,2A,C3,D5, 2232
356 DATA 9C,7A,1E,4B,0F,0F,0F,0F,B3,57,3E, [2980]
ED,C3,39,9D,67,7A,FE,03,C2, 2093
357 DATA E0,9E,7C,CD,B6,9E,20,05,3E,F9,C3, [3303]
29,9D,FE,05,C2,E0,9E,06,F9, 2882
358 DATA C3,42,9D,FE,0B,20,39,CD,EC,9E,CD, [2196]
90,9E,20,09,3E,32,ED,4B,3F, 2406
359 DATA AB,C3,D5,9C,FE,04,20,15,78,FE,04, [4898]
CA,E0,9E,FE,02,ED,4B,3F,AB, 2810
360 DATA 1E,43,20,B0,3E,22,C3,D5,9C,FE,05, [3659]
C2,E0,9E,79,16,22,ED,4B,3F, 2352
361 DATA AB,C3,39,9D,FE,0C,20,18,78,FE,02, [3769]
D2,E0,9E,CD,EC,9E,CD,90,9E, 2976
362 DATA C2,E0,9E,ED,4B,3F,AB,3E,02,C3,18, [2292]
9D,FE,05,20,1C,CD,EC,9E,FE, 2734
363 DATA 03,20,08,3A,3F,AB,16,21,C3,39,9D, [3283]
FE,0B,C2,E0,9E,3A,3F,AB,16, 1954
364 DATA 2A,C3,39,9D,FE,06,20,1B,CD,EC,9E, [3796]
CD,90,9E,C2,E0,9E,3A,40,AB, 2745
365 DATA FE,01,20,04,06,4F,18,02,06,47,0E, [3010]
ED,C3,42,9D,FE,0D,20,26,CD, 1690
366 DATA EC,9E,FE,03,20,0F,16,36,AF,B8,DA, [3753]
E0,9E,3A,3F,AB,41,4F,C3,36, 2418
367 DATA 9D,FE,01,C2,E0,9E,78,F6,70,4F,3A, [1952]
3F,AB,47,C3,47,9D,FE,01,C2, 2780
368 DATA E0,9E,3A,40,AB,07,07,07,32,40,AB, [3824]
CD,EC,9E,5F,3A,40,AB,57,FE, 2309
369 DATA 38,20,2E,7B,FE,0B,20,05,3E,3A,C3, [2902]
D5,9C,FE,0C,20,0B,78,FE,02, 1928
370 DATA D2,E0,9E,3E,0A,C3,18,9D,FE,06,20, [3293]
11,0E,ED,78,FE,01,20,05,06, 2018
371 DATA 5F,C3,42,9D,06,57,C3,42,9D,3E,40, [3665]
B2,67,3E,06,B2,6F,3E,46,B2, 2098
372 DATA 57,7B,C3,DD,9C,21,6B,AB,06,14,7E, [4954]
CD,E0,9D,77,23,10,F8,3A,60, 2403
373 DATA AB,C3,3F,90,00, 573                [1093]
374 MEMORY &8EAF:dat=0:sz=0:dz= 10:CLS:PRI [3476]
NT"Bitte etwas Geduld"
375 FOR adr=-29008 TO-21744                 [2002]
376 READ b$:byte=VAL("&"+b$):dat=dat+1:sz= [4614]
sz+byte:POKE adr,byte
377 IF dat< 20 AND adr<-21744 THEN 380      [2336]
378 READ chksum:IF chksum <>sz THEN PRINT  [4000]
"Fehler in Zeile :";dz
379 dz=dz+ 1 :sz=0:dat=0                    [893]
380 NEXT                                    [350]
381 PRINT"Fertig. Wurde kein Fehler anzeig [5737]
t , so speichern Sie zuerst die Data-Zeile
n ab.":END
```



Erstellen Sie Ihre eigenen Grafik-Abenteuer

Mit dem flexiblen, leicht anzuwendenden Grafik-Adventure-Creator.



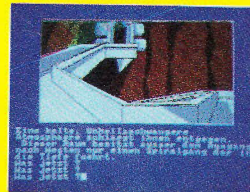
Mit dem GAC ist das Erzeugen von Grafiken, wie diesen, auf einfachste Art möglich.
Im Grafik-Editor sind Funktionen wie Line, Dot, Circle, Ellypse und





Fast Fill enthalten. Es ist möglich, Bilder miteinander zu kombinieren oder in einem Bild Schritt für Schritt vor- und zurückzublättern.
Weiterhin ist es möglich,

765 Verben und Substantive in ein Abenteuer einzubauen, alphabetisch zu überprüfen und zu sortieren.



Flexible Textgestaltung
erlaubt Ihnen umfangreiche Beschreibungen Ihrer Lokationen und das Erstellen Ihrer eigenen Kommentare. GAC formiert diese Kommentare und Texte nach Textverarbeitungsprinzip, so daß Ihr eigenes Adventure "professionell" aussieht.

Der "Graphik-Adventure-Creator" wurde speziell für die Schneider CPC Rechner entwickelt, um es Ihnen zu ermöglichen, schnelle und kompakte Grafik-Abenteuerspiele mit voller Ausnutzung der Rechnermöglichkeiten und einem minimalen Aufwand an Programmierkenntnissen zu erstellen.

Kompletter, intelligenter Kommando-Interpreter (Parser)
Der Parser versteht komplexe Sätze und Mehrfach–Kommandos. Sie werden nicht auf Zwei-Wort-Kommandos wie "geh Nord" beschränkt. GAC "versteht" auch Eingaben wie z.B. "nimm die Truhe und öffne sie vorsichtig. Untersuche das Medallion und lege es in die Truhe."

Der Parser erkennt bis zu fünf Buchstaben eines Kommandos und kann auch ähnlich klingende Worte unterscheiden.

P.S.:
Abenteuer, die Sie mit GAC erstellt haben, laufen völlig unabhängig vom Creator. Das heißt, daß Sie Ihre Abenteuer verschenken oder vermarkten können.

GAC-Abenteuer können einfach mit Run gestartet werden, also ohne den eigentlichen Creator zu benötigen. Außerdem enthält der GAC eine komplette deutsche Bedienungsanleitung.

Geld verdienen mit dem GAC

Mit diesem Programm ist es möglich, Abenteuer professioneller Qualität zu entwickeln. Programme, die mit dem GAC erstellt wurden, laufen ohne den Creator "Stand allone". Sie können Ihre Programme also professionell vermarkten und verkaufen. Lassen Sie Ihrer Phantasie freien Lauf mit dem Graphic-Adventure-Creator.

Wenn Sie ein spannendes, lustiges oder gruseliges Abenteuer mit dem GAC geschrieben haben, lassen Sie es uns wissen.

Vertrieb in Deutschland, Österreich und der Schweiz über: DMV-Verlag -Leserservice-, Fuldaer Str. 6, 3440 Eschwege, Telefon: (0 56 51) 87 02, Telex 993 210 dmv d

Bitte benutzen Sie die Bestellkarte!

Händleranfragen willkommen!

Der Preisbrecher in aller Kürze

Schneider PC:



In letzter Minute (der Layouter war hocherfreut) haben wir noch Platz für diesen Artikel im Heft angemeldet. Der Grund allerdings rechtfertigt die aufgetretene Hektik: Die Gerüchteküche hat (bis auf weiteres) ausgekocht; der mit Spannung erwartete Schneider PC ist „aus dem Sack".

Leider sind die zur Zeit zur Verfügung stehenden Informationen recht knapp, so daß Sie sich zunächst mit einigen Fotos und Auszügen aus einem Presseblatt, gemischt mit ersten eigenen Eindrücken, begnügen müssen. Heft 10 der CPC International wird jedoch eine ausführliche Soft- und Hardwarebeschreibung präsentieren.

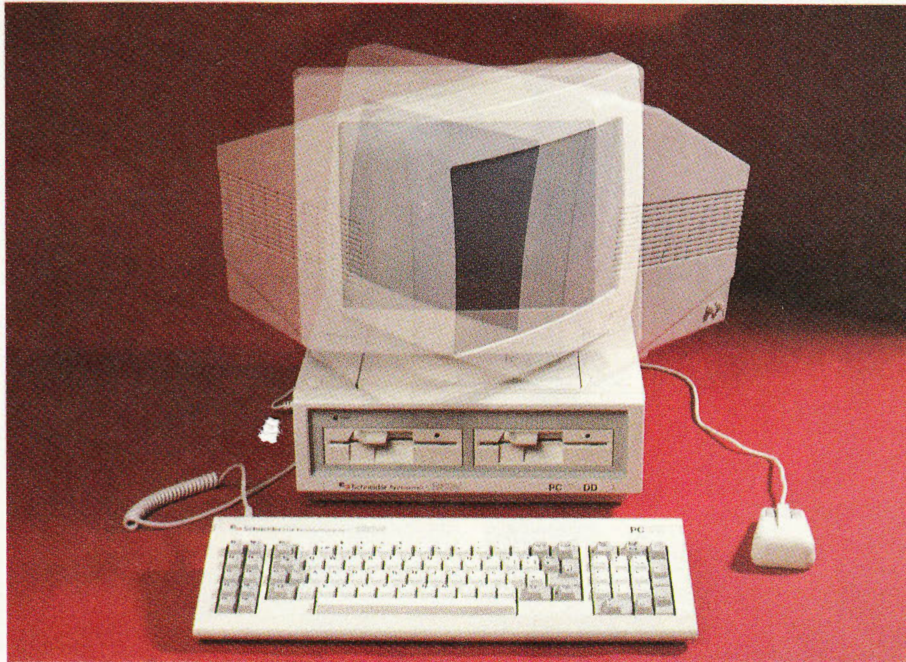
Die Hardware

Der Schneider PC ist, wie die meisten Geräte der PC-Klasse, dreigeteilt. Der Monitor (wahlweise monochrom 'grey level' – oder als Farbmonitor erhältlich), enthält auch die komplette Stromversorgung des Rechners. Ein sehr nützliches Feature ist der sehr gelenkige Schwenkfuß des Monitors (Bild 3), der sich den Bedürfnissen des Bedieners gut anpaßt. Der Monitor kann in eine passende Mulde im Zentralgehäuse gestellt werden, ebenso ist eine Aufstellung neben diesem, dank ausreichend langer Kabel, möglich.

Das Zentralgehäuse enthält die eigentlich interessanten Details des neuen Rechners, wie die Prozessorplatine, die Schnittstellen und in der Standardversion ein 5,25"-Laufwerk. Die in Bild 2 gezeigte, uns zur Ansicht zur Verfügung gestellte Version war schon mit zwei Laufwerken bestückt; die Standardversion enthält anstatt des linken Laufwerkes eine Blindplatte. Die doppelköpfigen Laufwerke bieten eine maximale Speicherkapazität von je 360 kB; zusätzlich kann über einen der drei auf der Hauptplatine vorhandenen Slots eine Harddisk mit Controller angeschlossen werden.

Auf der Geräterückseite sind zwei 25-polige, sogenannte Min.-D-Sub-Steckverbindungen vorhanden, über die die von je einer seriellen und einer parallelen Schnittstelle erzeugten Signale zu ihren Adressaten gelangen.

Das Herz des Rechners ist die bekannte 8086 CPU, die im Schneider PC mit 8 MHz-Takt getrieben wird. Direkt neben der CPU wartet ein leerer Sockel auf den Arithmetik-Coprozessor 8087, der eine um ein Vielfaches erhöhte Rechengeschwin-



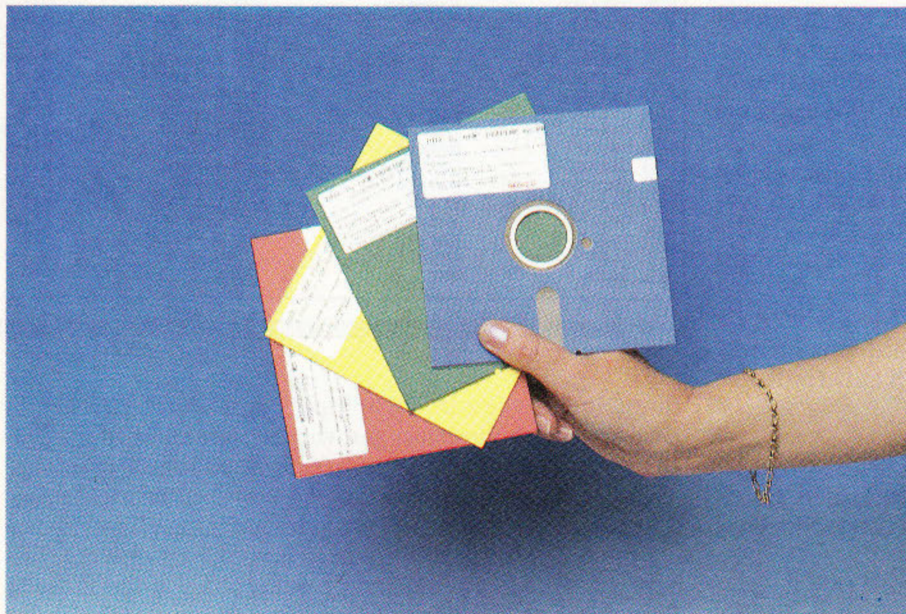
Der Monitor ist sehr anpassungsfähig

Die Tastatur ist in der Anordnung der Tasten der IBM-PC-Tastatur nachempfunden; zwei Füße am hinteren Ende des Keyboards erlauben eine leichte Vorwärtsneigung.
Last, not least, steht dem User eine Maus mit zwei Tasten zur komfortablen Bedienung des Desktops und verschiedener Anwenderprogramme zur Verfügung, sogar der obligatorische 9-polige Joystickport ist vorhanden.

Die Software...

wird auf vier Disketten bereitgestellt (Bild 4) und einfach der Farbe der Diskettenhüllen nach gebootet (rot, blau, grün) - eine nützliche Hilfe für den Anwender.
Der Schneider PC kann unter den mitgelieferten Betriebssystemen MSDOS 3.2 und DOS Plus arbeiten, wodurch eine weitgehende Kompatibilität zu CP/M 86 und PC-DOS erreicht wird. Diese Betriebssysteme unterstützen die hardwaremäßige Netzwerkfähigkeit des Schneider PC, ebenso ist laut Herstellerangaben ein Multitasking möglich.
Auf einer weiteren Diskette wird die von Digital Research entwickelte GEM-Software bereitgestellt, die die Maus und ein sogenanntes "Icon-Operating-System" verwaltet. GEM-Desktop stellt eine mausbetriebene Verwaltung der Computeroperationen mittels Pull-down-Menues zur Verfügung, leider können nur maximal drei Fenster geöffnet werden. Die GEM-Software kann jedoch vom selbstprogrammierenden Anwender voll genutzt werden, so auch GEM-Paint, mit dem in Zusammenarbeit mit der Grafikkarte komfortable Farbgrafiken erstellt werden können.
Locomotive Basic 2 schließlich ist ein neuer Basic-Interpreter, der strukturierte Programmierung durch Wegfall der Zeilennummern und namentlich aufrufbare Programmodule erleichtert.
Leider ist hier schon das Ende der Aufzählung; auf die Beigabe nützlicher Anwendersoftware wie etwa einer Textverarbeitung und einer Dateiverwaltung wurde (wohl aus Preisgründen) verzichtet.

digkeit und -genauigkeit ermöglicht. Dieser ist im freien Handel in der 8 MHz-Version für etwa 450,- DM zu haben, kann seine Leistungsfähigkeit aber nur in Zusammenarbeit mit passender Software unter Beweis stellen.
Ein weiterer erfreulicher Umstand ist das Vorhandensein einer Farb-Grafikkarte, die schon in der Standardversion verfügbar ist und eine 16-Farb-Darstellung bei einer Auflösung von 640 x 200 Punkten ermöglicht.
Weiterhin ist eine batteriegepufferte Hardware-Uhr vorhanden, die jederzeit vom Desktop aus bedient werden kann; die Batterien versorgen gleichzeitig einen nicht flüchtigen RAM-Block, in dem der Benutzer die gewünschte Einschaltkonfiguration ablegen kann.



Schön bunt - die mitgelieferte Software ist farblich gekennzeichnet.

Zusammenfassung der Daten

- wahlweise Monochrom- oder Farbmonitor verfügbar
- Textbildschirm 16 Farben, 40 x 25 Zeichen oder 80 x 25 Zeichen
- Grafikbildschirm: drei Vierfarbpaletten, 320 x 200 Punkte oder zweifarbig, 640 x 200 Punkte oder 16-farbig, 640 x 200 Punkte
- Prozessor 8086/8MHz, Sockel für Coprozessor 8087 vorhanden
- 512 kB Speicher, auf max. 640 kB auf der Hauptplatine aufrüstbar
- wahlweise ein oder zwei 5,25"-Doppelkopflaufwerke je 360 kB
- batteriegepufferte Hardwareuhr
- batteriegepuffertes Anwender-RAM
- je eine Standard-Schnittstelle seriell und parallel
- drei PC-Slots
- IBM-PC-kompatible Tastatur
- Maus mit eigenem Port
- Joystickport
- Software: GEM, GEM-Desktop, GEM-Paint (Digital Research);
- DOS Plus (Digital Research);
- MSDOS (Microsoft)
- Locomotive Basic 2
- Preis der Standardversion (Monochrom-Monitor, ein Laufwerk, 512 kB): unter DM 2000,- (ME)



Listenverarbei

Häufig werden Listen aller Art benötigt: Kegellisten, Kaffeelisten, Klassenlisten, Vereinslisten, Teilnehmerlisten usw. Es gibt zwar jede Menge Sortierroutinen aller Art, aber sie allein bilden noch kein komfortables Listen-Programm.
Die veröffentlichten Sortierroutinen sind in der Regel so allgemein gehalten, daß sie zwar das Prinzip des Sortierens verdeutlichen, jedoch nicht für den Anwender unmittelbar brauchbar sind.
Weiterhin haben sie alle den entscheidenden Fehler, daß sie bei dem Sortiervorgang die Umlaute im "deutschen" Sinne nicht berücksichtigen. Sie werden - wegen der ASCII - noch hinter "z" eingeordnet.

Deshalb habe ich ein Programm entwickelt, das für dieses Problem eine - so hoffe ich - brauchbare Lösung bietet. Ein "ä" z.B. wird beim Sortieren als "ae" behandelt und auch so einsortiert. Analog dazu erfolgt die Sortierung bei den anderen Umlauten.
Das Programm wurde für die Erstellung von Klassenlisten entwickelt. Daher ist die Anzahl der zu erfassenden Namen auf maximal 42 beschränkt. Im Prinzip könnte die Anzahl der Namen erhöht werden, lediglich die Bildschirmausgabe müßte abgeändert werden.
In einem Menue wird dem Anwender die Möglichkeit eröffnet, auch Überschriften nach freier Wahl einzugeben. So wird eine Nutzung des Programms auch außerhalb des Schulbereichs ermöglicht.

Wer eine Festeinstellung auf "freie Überschrift" wünscht, braucht nur die Remarks in 265 und 3465 entfernen. Es erscheinen dann nur noch die auf die "freien Überschriften" zugeschnittenen Menues.

Das Programm ist sofort lauffähig auf dem 664 und 6128. Wegen einiger Befehle, die in dem Programm Verwendung finden, wie z.B. "clear input" und "derr", müssen für den 464 einige Streichungen vorgenommen werden. Dies hat keine wesentlichen Auswirkungen auf den Programmablauf. Man sollte nur bei kleinen Wartezeiten nicht ungeduldig auf irgendwelche Tasten hauen. Diese Tastenbetätigungen werden nämlich intern gespeichert und erscheinen dann als unliebsame, scheinbar unerklärliche Eingaben auf dem Bildschirm. Bei den neueren CPC's werden diese Tastendrücke durch "clear input" abgefan-

1.	Acar	Ilhan	22.	Kolodziej	Anja
2.	Äbli	Heinz	23.	Lucks	Nicole
3.	Aladag	Nuri	24.	Munzert	Gerd
4.	Arduc	Özkan	25.	Oebel	Klaus
5.	Atalay	Mehmet	26.	Oruc	Sennur
6.	Bänsch	Manuela	27.	Oßau	Claudia
7.	Bamberg	Chris	28.	Pürcü	Melek
8.	Basiyigit	Mehmet	29.	Pürcü	Sabahattin
9.	Brockkötter	Brigitte	30.	Rombowsky	Michaela
10.	Bushoven	Wolf	31.	Schimmelfeder	Bodo
11.	Catak	Yahya	32.	Schmidt	Nicole
12.	Cinar	Rayif	33.	Schmidt	Olaf
13.	Dikec	Fatma	34.	Schulz	Dieter
14.	Dogru	Mehmet	35.	Schulz	Elisabeth
15.	Durmaz	Gülten	36.	Sprung	Klaus Uwe
16.	Eyriyer	Hatice	37.	Tewes	Wolfgang
17.	Gabryschak	Frank	38.	Ustabas	Reyhan
18.	Huwald	Heinz	39.	Winter	Stephan
19.	Huwald	Karola			
20.	Klos-Kunze	Mechthild			
21.	Koch	Uwe			

(A)bspeichern (L)aden (V)eränderung (E)nde (N)eu (D)ru



gen. Eine nähere Erklärung zu den nötigen Änderungen erfolgt weiter unten.

Die Druckroutine wurde für den NLQ 401 geschrieben.

Leistungen des Programms

- geeignet für Listen bis zu 42 Namen
- Alle Eingaben erfolgen in Kleinschrift. Das lästige <shift>-Drücken bei den Anfangsbuchstaben entfällt daher. Eine Umwandlung erfolgt automatisch.
- Eine bündige Ausgabe der Nach- und Vornamen erfolgt automatisch. Bei der Eingabe braucht keine Rücksicht auf diese Formatierung genommen werden.
- Umlaute dürfen original eingegeben werden.
- Sortieren der Namen. Umlaute werden nach "deutscher Norm" einsortiert.
- Abspeichern und Laden der Listen.
- Löschen und Hinzufügen von Namen jederzeit möglich. Der Löschvorgang ist dabei auf dem Bildschirm sichtbar.
- Drucken mit oder ohne Vornamen.
- Beliebig viele Kopien möglich.
- Freie Wahl der Überschriften auf der Druckliste.
- Trennstriche zwischen den Namen. Daher besonders gut geeignet für Eintragungen aller Art.

Beschreibung des Programms

Nach dem Starten des Programms erscheint der Name des Programms und nach kurzer Wartezeit die Wahl der Überschrift (Bild Nr. 1).

```
 Wahl der überschriften
  auf der Druck-Liste

      Klassenliste..
      Pflegschaft...
      Freie Wahl....

 Bitte passende Taste drücken !
```

Bild Nr. 1

Drückt man z.B "F" für "freie Überschrift", so kann man den eigenen Namen, die Überschrift der Liste und das Datum der Ausgabe der Liste eingeben. Wenn man eine von den drei Möglichkeiten nicht ausfüllen will, kann man <enter> drücken (Bild Nr. 2)!

```
 Wahl der überschriften
  auf der Druck-Liste

 Name.......:wolfgang tewes
 überschrift:kegelabend
 Datum......:29.5.86

 Die Eingabe mit <enter> abschließen
```

Bild Nr. 2

Alle Eingaben – auch die der folgenden Programmteile – erfolgen in Kleinschrift. Eine Routine im Programm sorgt für die richtige Umwandlung in Großbuchstaben an den passenden Stellen. Der Computer muß sich also vorher in dem Kleinschrift-Modus befinden! Dies ist aber wohl normalerweise der Fall.

Man kann nun zwischen den Möglichkeiten "alte Liste" oder "neue Liste" wählen (Bild Nr. 3)

```
 Alte Liste von Disc
 Neue Liste erstellen

 Bitte passende Taste drücken !
```

Bild Nr. 3

Beim Drücken von "A" für "alte Liste" kann man eine schon abgespeicherte Liste von Disk (Kassette) einlesen. Beim Drücken von "N" für "neue Liste" erscheint die Bildschirm-Maske für die Eingabe der Liste (Bild Nr. 4). Die Eingabe geschieht im folgenden Format: nachname,-vorname. Das Programm ist so geschrieben, daß es einige Eingabefehler abfängt. Z.B. werden die Eingaben "Schulz, Heinrich" oder "schulz, Heinrich" anstandslos übernommen und richtig verarbeitet.

ANWENDUNG

Nach der Eingabe werden die Namen, formatiert nach den jeweils längsten Namen, auf dem Bildschirm in einem speziellen Window ausgegeben. Eine Lücke von drei Leerzeichen hebt den Nach- und Vornamen optisch voneinander ab. Beim Druck später werden die Namen nach dem längsten Nachnamen ausgerichtet. Man kann also unbekümmert die Namen – so wie sie kommen – eingeben. Für eine säuberliche Auflistung sorgt das Programm!

Zum Schluß gibt man den Pfeil <:> so ein, als ob er ein Name wäre.

```
Keine Großbuchstaben benutzen!
EINGABE-FORMAT:   nachname,vorname

1. Höttgens   Erna
2. Abli       Heinz
3. Huwald     Karola

? huwald,karola
```

Bild Nr. 4

Umlaute können eingegeben werden. Dabei gilt folgende Tastenbelegung: @-> ß, [-> a,] -> ö, /-> ü. Da man ja im Kleinschrift-Modus arbeitet, werden nur die "Klein-Umlaute" benötigt.
Nach der Eingabe des Pfeils und <enter> wird die Liste geordnet.

Zu dem Sortiervorgang

Wegen der ASCII-Norm würden Namen mit Umlauten falsch eingeordnet. Um dieses Problem in den Griff zu bekommen, muß man zu einem kleinen Trick greifen. Er besteht darin, daß vor dem eigentlichen Sortiervorgang z.B der Umlaut "ä" in "ae" umgewandelt wird. Gleichzeitig bekommt der Name, der diesen Umlaut enthält, die Feldnummer als Kennzahl ans Ende angehängt. Vorher wird der ursprüngliche Name in ein anderes Feld mit der gleichen Feldnummer umgeschaufelt. Gleiches geschieht mit den anderen Umlauten.

Nun findet mit diesem veränderten Namen ein Sortiervorgang statt. Nach dem Sortieren werden die ursprünglichen Namen mit Umlauten – unter Verwendung der Kennzahlen – wieder in das alte Feld zurückgeholt. Die "Anhängsel" an den Namen werden entfernt.
Um die Zeit der Schleifendurchgänge zu verkürzen, werden die Namen, die einen Umlaut beinhalten, schon bei der Eingabe der Namen mit einem "*" am Ende versehen. Dadurch erreicht man, daß nicht alle Namen auf Umlaute hin durchsucht werden müssen. Die Wartezeit beim Sortieren wird dadurch wesentlich verkürzt.

Dies hat aber den Nachteil, daß der Programmieraufwand erhöht wird, insofern daß vor allen Bildschirm- und Druckerausgaben diese Sterne entfernt werden müssen.
Auf dem Bildschirm erscheint nun die geordnete Liste. Nach all dem Mühsal, die der Computer mit den Umlauten hatte, übrigens in beachtenswerter Geschwindigkeit! In der untersten Zeile erscheint das Hauptmenue, das mannigfaltige Auswahl zuläßt. Z.B. ist hier Laden und Abspeichern der Liste möglich (Bild Nr. 5).

```
 1. Acar         Ilhan        22. Kolodziej      Anja
 2. Abli         Heinz        23. Lucks          Nicole
 3. Aladag       Nuri         24. Munzert        Gerd
 4. Arduc        Özkan        25. Oebel          Klaus
 5. Atalay       Mehmet       26. Oruc           Sennur
 6. Bänsch       Manuela      27. Oßau           Claudia
 7. Bamberg      Chris        28. Pürcü          Melek
 8. Basiyigit    Mehmet       29. Pürcü          Sabahattin
 9. Brockkötter  Brigitte     30. Rombowsky      Michaela
10. Bushoven     Wolf         31. Schimmelfeder  Bodo
11. Catak        Yahya        32. Schmidt        Nicole
12. Cinar        Rayif        33. Schmidt        Olaf
13. Dikec        Fatma        34. Schulz         Dieter
14. Dogru        Mehmet       35. Schulz         Elisabeth
15. Durmaz       Gülten       36. Sprung         Klaus Uwe
16. Eyriyer      Hatice       37. Tewes          Wolfgang
17. Gabryschak   Frank        38. Ustabas        Reyhan
18. Huwald       Heinz        39. Winter         Stephan
19. Huwald       Karola
20. Klos-Kunze   Mechthild
21. Koch         Uwe

(A)bspeichern  (L)aden  (V)eränderung  (E)nde  (N)eu  (D)rucken
```

Bild Nr. 5

Eine der interessantesten Möglichkeiten, ist das "Verändern". Drückt man "V" so erscheint ein Untermenue:

Löschen – Hinzufügen – Beides.

Wählt man eine der drei Möglichkeiten, so wird man in einem Window in den untersten Zeilen durch Abfragen geführt.

Bei der Wahl "Hinzufügen" erscheint in einem Window der letzte Name der aktuellen Liste. Der weitere Ablauf gestaltet sich genauso wie bei der Eingabe der Namen. Dabei werden die Namen bündig zu dem letzten Namen ausgegeben. Ist jedoch der neue Nachname länger als der längste, bisher vorgekommene Name, so werden die nun folgenden Namen nach diesem neuen Format ausgerichtet. Die gesamte Liste wird – für den Anwender unmerklich – umformatiert, so daß bei den späteren Bildschirm- bzw. Druckausgaben wieder alles im Lot ist.
Bei der Wahl "Löschen" wird nach Eingabe des zu löschenden Namens und nach einer Sicherheitsabfrage ein breiter Balken über den betreffenden Namen gezogen (Bild Nr. 6).

```
 1. Acar         Ilhan        22. Kolodziej      Anja
 2. Abli         Heinz        23. Lucks          Nicole
 3. Aladag       Nuri         24. Munzert        Gerd
 4. Arduc        Özkan        25. Oebel          Klaus
 5. Atalay       Mehmet       26. Oruc           Sennur
 6. Bänsch       Manuela      27. Oßau           Claudia
 7. Bamberg      Chris        28. Pürcü          Melek
 8. Basiyigit    Mehmet       29. Pürcü          Sabahattin
 9. Brockkötter  Brigitte     30. Rombowsky      Michaela
[redacted]                    31. Schimmelfeder  Bodo
11. Catak        Yahya        32. Schmidt        Nicole
12. Cinar        Rayif        33. Schmidt        Olaf
13. Dikec        Fatma        34. Schulz         Dieter
14. Dogru        Mehmet       35. Schulz         Elisabeth
15. Durmaz       Gülten       36. Sprung         Klaus Uwe
16. Eyriyer      Hatice       37. Tewes          Wolfgang
17. Gabryschak   Frank        38. Ustabas        Reyhan
18. Huwald       Heinz        39. Winter         Stephan
19. Huwald       Karola
20. Klos-Kunze   Mechthild
21. Koch         Uwe

10 . Bushoven    Wolf     richtig ? (J/N)
```

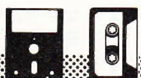
Bild Nr. 6

In allen Fällen erscheint am Schluß wieder die Liste in Mode 2 mit dem Hauptmenue.
Nach der Wahl "Drucken" erscheint das Menue der Druckparameter, welches die Auswahl zwischen folgenden Optionen zuläßt:
Mit/ohne Vornamen, Anzahl der Kopien, Schönschrift/Normalschrift.
Die Normalschrift ist dabei so eine Art Schnelldruck. Der Umgang mit diesem Menue wird im unteren Window erklärt (Bild Nr. 7).
Nach dem Druck hat man dann die Möglichkeit, nochmals zu drucken, zu beenden oder ins Hauptmenue zurückzukehren.
Bei "Neu-Druck" kann man per Menue die Überschriften und die Druck-Parameter verändern. Die Liste bleibt dabei unverändert. Auch hier wird das weitere Vorgehen

durch einen erklärenden Text im unteren Window erläutert.
Bleibt noch das Einlesen von schon abgespeicherten Listen zu erklären. Zu Beginn des Programms oder per Haupt-Menue hat man ja diese Wahl.

Im oberen Teil des Bildschirms erscheinen die aktuellen Dateien der eingeschobenen Disk - und zwar nur diese! Andere Einträge dieser Disk werden nicht angezeigt. Leider sind sie nicht alphabetisch geordnet. Das liegt an dem CP/M-Befehl "dir". Die Alternative "cat", die bekanntlich alphabetisch sortiert, läßt die Beschränkung "*.dat" nicht zu.

Im unteren Window gibt man den Namen der aufzurufenden Datei ein. Nach <enter> wird sie geladen und erscheint auf dem Bildschirm. Die weitere Verarbeitung erfolgt dann per Menue.



Anpassung an den CPC 464

Alle "clear input" müssen entfernt werden. Die Zeile 4370 muß gelöscht werden. Die Zeile 200 muß lauten: On error goto 4380.

1. 464 mit Kassetten-Recorder
Folgende Zeilen müssen gelöscht, bzw. verbessert werden:

Delete 1530 - 1570 / Delete 1690 - 1750
1580 openin " / 1760 openout "
Die Kassette muß schon auf den Anfang der zu ladenden Datei eingestellt sein.

2. 464 mit Diskettenstation
Die Zeilen 1540 und 1700 müssen verändert werden.
1540 a1$="*.dat":ödir,@1$:print st1$
1700 print st2$:a1$="*.dat":ödir,@1$:print st2$
Das "ö" vor dem Dir ist der senkrechte Strich <shift> < >.

Bemerkungen
Das Programm ist auf dem CPC 664 sofort ohne Änderungen lauffähig. Es wird gestartet mit **run "listenlq"**.

(Wolfgang Tebes)

```
1 '*********************************        [1460]
2 '********        LISTE         ********    [1068]
3 '******** Vers. 2.0 NLQ 401 ********       [1163]
4 '*********************************        [1460]
5 '                                          [117]
6 '********   Wolfgang Tewes   ********      [1634]
9 '                                          [117]
11 '*********************************       [1391]
12 '                                         [117]
13 '**** Vers. vom 21.5.86 fuer NLQ 401 ** [2079]
****
20 GOSUB 4190:GOSUB 4120:GOTO 190           [1999]
30 m$=UPPER$(INKEY$):IF m$="" THEN 30 ELSE [1498]
 RETURN
40 '                                        [117]
50 '****************** Initialisierung M [2748]
ode 1 ****************
60 '                                        [117]
70 MODE 1:INK 0,1:INK 1,24:INK 2,0:INK 3,6 [12680]
:PAPER 0:PEN 1:BORDER 1:WINDOW #1,1,40,1,3
:WINDOW #2,4,37,5,21:WINDOW #3,1,40,23,25:
PAPER#1,1:PAPER#2,2:PAPER#3,3:PEN#1,3:PEN#
2,1:PEN#3,2:RETURN
```

```
Wahl der überschriften
auf der Druck-Liste

Klassenliste..
Pflegschaft...
Freie Wahl....

Bitte passende Taste drücken !
```

```
80 '                                        [117]
90 '***************** Umrandung Window 2 ** [2827]
*********************
100 '                                       [117]
110 en=0:x1=46:y2=338:x2=592:y1=62:GOSUB 2 [3386]
700:RETURN
120 '                                       [117]
130 '************** Meldung Window 3 ***** [2614]
***************
140 '                                       [117]
150 CLS#3:LOCATE #3,7,2:PRINT#3,"Bitte pas [4659]
sende Taste dr]cken !":RETURN
160 '                                       [117]
170 '******************** Beginn Programm [3031]
*****************
180 '                                       [117]
190 CLEAR INPUT:CLS:DEFINT a-z:DIM a$(50), [1858]
b$(50)
200 ON ERROR GOTO 4370                     [1344]
210 GOSUB 70:CLS                           [1170]
220 '                                       [117]
230 '****** MEN} }BERSCHRIFTEN ******       [1211]
240 '                                       [117]
250 DATA "Name.......:","Klasse.....:","Fa [4857]
ch.......:","Datum......:","}berschrift:"
260 FOR m=1 TO 5:READ x$(m):NEXT           [1421]
265 'm$="F":GOTO 310                        [549]
270 GOSUB 1410:'Welche ]berschrift          [2567]
280 GOSUB 30                                [852]
290 IF m$="K" THEN uber$="Klassenliste":GO [3932]
TO 330
300 IF m$="P" THEN uber$="Klassenpflegscha [4280]
ft":x$(3)=x$(4):GOTO 330
310 IF m$="F" THEN x$(2)=x$(5):x$(3)=x$(4) [2881]
:freie=1:GOTO 330
320 GOTO 280                                [413]
330 CLS#2:z=7:FOR m=1 TO 3:LOCATE#2,2,z:z= [5957]
z+2:PRINT#2,x$(m):NEXT:GOSUB 4080
340 z=7:FOR m=1 TO 3:LOCATE#2,14,z:z=z+2:C [5907]
LEAR INPUT:LINE INPUT#2,"",ein$(m)
350 IF ein$(m)="" OR LEN(ein$(m))=1 THEN 3 [2306]
70
360 m$=ein$(m):GOSUB 890:ein$(m)=m$        [2023]
370 NEXT                                    [350]
380 '                                       [117]
390 '******************* Men] alte oder neu [2416]
e Liste ****************
400 '                                       [117]
410 CLS#1:CLS#2:LOCATE #2,8,8:PEN#2,3:PRIN [10406]
T#2,"A";:PEN#2,1:PRINT#2,"lte Liste von Di
sc":LOCATE #2,8,10:PEN#2,3:PRINT#2,"N";:PE
N#2,1:PRINT#2,"eue Liste erstellen":GOSUB
150
420 GOSUB 30                                [852]
430 IF m$="A" THEN 1500                     [743]
440 '                                       [117]
450 '****** Eingabe der Namen ******        [2581]
460 '                                       [117]
470 CLS#1:PRINT#1,TAB(3)"K e i n e  Gro~bu [4687]
chstaben benutzen!"
480 PRINT#1                                 [318]
490 PRINT#1,TAB(4)"EINGABE-FORMAT:  nachna [3144]
me,vorname"
500 KEY DEF 31,0,44,44                     [1001]
510 i=1:n=1:flag=1                         [1021]
520 CLS#2                                   [372]
530 WHILE a$(i-1)<>"^"                     [1118]
540 CLS#3:PRINT CHR$(7):LOCATE#3,1,2:CLEAR [4054]
 INPUT:LINE INPUT#3,"";a$(i)
550 IF flag=1 THEN GOSUB 660 ELSE GOSUB 73 [1975]
0
```

Wahl der Überschriften
auf der Druck-Liste

Name.......:wolfgang tewes
Überschrift:kegelabend
Datum......:29.5.86

Die Eingabe mit <enter> abschließen

```
560 PEN#2,3:PRINT#2,USING "##";i;:PRINT#2,  [4461]
". ";:PEN#2,1:PRINT#2,LEFT$(a$(i),29)
570 IF um=1 THEN a$(i)=a$(i)+"*":um=0:'***  [4112]
* Markieren falls Umlaut
580 i=i+1:n=n+1                              [659]
590 WEND                                     [390]
600 KEY DEF 31,1,46,46                       [719]
610 a$(i-1)="":i=i-2:n=n-2                   [967]
620 GOTO 1020                                [351]
630 '                                        [117]
640 '********* Ermitteln der L)cke im erst   [3335]
en Namen ********
650 '                                        [117]
660 CLS#1:PRINT#1,TAB(15)"Die Taste < ^ >"   [6389]
:PRINT#1:PRINT#1,TAB(3)"beendet als letzte
 Eingabe die Liste."
670 m$=a$(i):GOSUB 890:a$(i)=m$              [1080]
680 p=INSTR(a$(i),","):IF p<>0 THEN ref=p-   [4947]
1::a$(i)=MID$(a$(i),1,p-1)+"   "+MID$(a$(i
),p+1):lang=ref+3:p=0
690 flag=0:RETURN                            [999]
700 '                                        [117]
710 '********** Ermitteln der L)cken der a   [3132]
nderen Namen *******
720 '                                        [117]
730 m$=a$(i):GOSUB 890:a$(i)=m$              [1080]
740 p=INSTR(a$(i),","):IF p=0 THEN 780       [1564]
750 IF p<>0 THEN kom=p-1:neu=kom+3           [1614]
760 IF neu>lang THEN ers=lang:lang=neu:ku=   [3360]
i-1:GOSUB 820
770 a$(i)=MID$(a$(i),1,p-1)+SPACE$(lang-ko   [1954]
m)+MID$(a$(i),p+1):p=0
780 RETURN                                   [555]
790  '                                       [117]
800  '************** Neuordnen falls l{nge   [4083]
rer Name ************
810  '                                       [117]
820  FOR m=1 TO ku                           [909]
830  a$(m)=LEFT$(a$(m),ers)+SPACE$(lang-er   [4362]
s)+MID$(a$(m),ers+1)
840  NEXT                                    [350]
850  RETURN                                  [555]
860 '                                        [117]
870 '******* Umwandlung in Gro~buchstaben    [2762]
************
880 '                                        [117]
890 FOR k=1 TO LEN(m$)-1                     [1436]
900 zei=ASC(MID$(m$,k+1,1)):z$=MID$(m$,k,1   [3094]
):uml=ASC(MID$(m$,k,1))
910 IF k=1 THEN IF(uml>47 AND uml<91) THEN   [3143]
 950 ELSE MID$(m$,k,1)=CHR$(uml-32):GOTO 9
50
920 IF z$="," AND MID$(m$,k+1,1)=" " THEN    [3025]
PO=k:le=1:GOTO 950
930 IF z$="," THEN IF zei>47 THEN IF zei<9   [5597]
1 THEN 950 ELSE MID$(m$,k+1,1)=CHR$(zei-32
):GOTO 950
940 IF z$=" " OR z$="-" OR z$="/" THEN IF    [5831]
zei>47 THEN IF zei<91 THEN 950 ELSE MID$(m
$,k+1,1)=CHR$(zei-32)
950 IF (uml>90 AND uml<94) OR (uml>122 AND   [3318]
 uml<127) THEN um=1
960 NEXT                                     [350]
970 IF le=1 THEN m$=MID$(m$,1,PO)+MID$(m$,   [1261]
PO+2):le=0
980 RETURN                                   [555]
990 '                                        [117]
1000 '********* Ver{nderung der Umlaute **   [3916]
*****
1010 '                                       [117]
1020 CLS#3:LOCATE#3,13,2:PRINT#3,"Liste wi   [3392]
rd geordnet."
1030 FOR i=1 TO n                            [456]
1040 IF RIGHT$(a$(i),1)<>"*" THEN 1210       [1669]
1050 b$(i)=a$(i)                             [425]
1060 SU=1                                    [219]
1070 p1=INSTR(su,a$(i),"["):IF p1<>0 THEN    [4791]
a$(i)=MID$(a$(i),1,p1-1)+"Ae"+MID$(a$(i),p
1+1):su=p1+2:GOTO 1070
1080 SU=1                                    [219]
1090 p2=INSTR(su,a$(i),"\"):IF p2<>0 THEN    [6318]
a$(i)=MID$(a$(i),1,p2-1)+"Oe"+MID$(a$(i),p
2+1):su=p2+2:GOTO 1090
1100 SU=1                                    [219]
1110 p3=INSTR(su,a$(i),"]"):IF p3<>0 THEN    [6041]
a$(i)=MID$(a$(i),1,p3-1)+"Ue"+MID$(a$(i),p
3+1):su=p3+2:GOTO 1110
1120 SU=1                                    [219]
1130 p4=INSTR(su,a$(i),"{"):IF p4<>0 THEN    [5195]
a$(i)=MID$(a$(i),1,p4-1)+"ae"+MID$(a$(i),p
4+1):su=p4+2:GOTO 1130
1140 SU=1                                    [219]
1150 p5=INSTR(su,a$(i),"|"):IF p5<>0 THEN    [5410]
a$(i)=MID$(a$(i),1,p5-1)+"oe"+MID$(a$(i),p
5+1):su=p5+2:GOTO 1150
1160 SU=1                                    [219]
1170 p6=INSTR(su,a$(i),"}"):IF p6<>0 THEN    [5001]
a$(i)=MID$(a$(i),1,p6-1)+"ue"+MID$(a$(i),p
6+1):su=p6+2:GOTO 1170
1180 su=1                                    [219]
1190 p7=INSTR(su,a$(i),"~"):IF p7<>0 THEN    [6537]
a$(i)=MID$(a$(i),1,p7-1)+"ss"+MID$(a$(i),p
7+1):su=p7+2:GOTO 1190
1200 a$(i)=a$(i)+STR$(i)                     [858]
1210 NEXT                                    [350]
1220 '                                       [117]
1230 ' ********* Sortieren ********          [1990]
1240 '                                       [117]
1250 FOR i=1 TO n-1                          [452]
1260 IF a$(i)<a$(i+1) THEN 1310              [366]
1270 a$(0)=a$(i+1):c=i                       [1465]
1280 a$(c+1)=a$(c):c=c-1                     [917]
1290 IF a$(0)<a$(c) THEN 1280                [1509]
1300 a$(c+1)=a$(0)                           [637]
1310 NEXT                                    [350]
1320 '                                       [117]
1330 '************* R]CKWANDLUNG DER UMLAU    [2763]
TE *************
1340 '                                       [117]
1350 FOR I=1 TO N                            [456]
1360 p=INSTR(A$(I),"*"):IF p=0 THEN 1380     [927]
1370 zahl$=MID$(a$(i),p+1):a=VAL(zahl$):a$   [3263]
(i)=b$(a):p=0
1380 NEXT                                    [350]
1390 GOSUB 2370:GOTO 2480                    [975]
1400 '                                       [117]
1410 '****** Abfrage welche )berschrift **   [3329]
********
1420 '                                       [117]
1430 CLS#1:PRINT#1,TAB(11)"Wahl der )bersc   [6919]
hriften":PRINT#1,TAB(13)"auf der Druck-Lis
te"
1440 GOSUB 110                               [935]
1450 CLS#2:LOCATE#2,11,7:PEN#2,3:PRINT#2,"   [5740]
K";:PEN#2,1:PRINT#2,"lassenliste.."
1460 LOCATE#2,11,9:PEN#2,3:PRINT#2,"P";:PE   [4376]
N#2,1:PRINT#2,"flegschaft..."
1470 LOCATE#2,11,11:PEN#2,3:PRINT#2,"F";:P   [3639]
EN#2,1:PRINT#2,"reie Wahl...."
1480 GOSUB 150:RETURN                        [1321]
1490 '                                       [117]
1500 '****** LADEN VON DISK ******           [1125]
1510 '                                       [117]
1520 CLS                                     [91]
1530 ST1$=STRING$(39,CHR$(208))              [1164]
1540 |DIR,"*.dat":PRINT ST1$                 [2620]
1550 CLS#3:LOCATE #3,2,2:INPUT #3,"Name de   [4059]
r Datei < ohne .dat > ........";a1$
1560 IF RIGHT$(a1$,4)=".dat" THEN 1580       [1598]
1570 a1$=a1$+".dat"                          [1285]
1580 OPENIN a1$                              [601]
1590 CLS#3:LOCATE #3,3,2:PRINT#3,"Die Date   [4106]
i "CHR$(24)a1$CHR$(24)" wird geladen."
1600 INPUT #9,n                              [698]
1610 FOR i=1 TO n:LINE INPUT#9,a$(i):NEXT    [880]
1620 INPUT#9,lang                            [1265]
1630 CLOSEIN                                 [752]
1640 GOSUB 2370:GOTO 2480                    [975]
1650 '                                       [117]
1660 '******** Sichern auf Disk ******       [1343]
```

ANWENDUNG

```
1670 '                                              [117]
1680 CLS                                            [91]
1690 ST2$=STRING$(79,CHR$(131))                     [1273]
1700 PRINT ST2$:|DIR,"*.dat":PRINT ST2$             [2041]
1710 CLS#1                                          [373]
1720 LOCATE #1,2,2:PRINT#1,CHR$(24)" Unter [9079]
 welchem Namen abspeichern < ohne .dat >..
 "CHR$(24);:INPUT#1,a1$
1730 IF LEN(a1$)>8 THEN CLS#1:LOCATE#1,2,1 [10420]
:PRINT#1,"Bitte nur ";CHR$(24);"8";CHR$(24
);" Buchstaben f]r den Datei-Namen eingebe
n!";:GOTO 1720
1740 IF RIGHT$(a1$,4)=".dat" THEN 1760              [1642]
1750 a1$=a1$+".dat"                                 [1285]
1760 OPENOUT a1$                                    [804]
1770 CLS#1:LOCATE#1,2,2:PRINT#1,"Die Datei [4447]
 ";CHR$(24)a1$CHR$(24)" wird abgespeichert
."
1780 PRINT#9,n                                      [810]
1790 FOR i=1 TO n                                   [456]
1800 PRINT#9,a$(i)                                  [995]
1810 NEXT                                           [350]
1820 PRINT#9,lang                                   [1158]
1830 CLOSEOUT:GOSUB 2370:GOTO 2480                  [2363]
1840 '                                              [117]
1850 '******** Hinzuf]gen *******                   [1588]
1860 '                                              [117]
1870 GOSUB 70:GOSUB 110                             [1376]
1880 CLS#1:PRINT#1,TAB(6)"Gib den (die) ne [7689]
uen Namen ein":PRINT#1:PRINT#1,TAB(9)"^^^
beendet die Eingabe.";
1890 CLS#2                                          [372]
1900 i=i-1                                          [440]
1910 IF RIGHT$(a$(i),1)="*" THEN PRINT#2,U [4487]
SING"##";i;:PRINT#2,". ";LEFT$(a$(i),LEN(a
$(i))-1):GOTO 1930
1920 PRINT#2,USING"##";i;:PRINT#2,". ";a$( [1759]
i)
1930 i=i+1:n=n+1                                    [659]
1940 KEY DEF 31,0,44,44                             [1001]
1950 WHILE a$(i-1)<>"^"                             [1118]
1960 CLS#3:PRINT CHR$(7):LOCATE#3,1,2:CLEA [4054]
R INPUT:LINE INPUT#3,"";a$(i)
1970 GOSUB 730                                      [939]
1980 PEN#2,3:PRINT#2,USING "##";i;:PRINT#2 [4461]
,". ";:PEN#2,1:PRINT#2,LEFT$(a$(i),29)
1990 IF UM=1 THEN A$(I)=a$(i)+"*":um=0              [1851]
2000 i=i+1:n=n+1                                    [659]
2010 WEND                                           [390]
2020 KEY DEF 31,1,46,46                             [719]
2030 a$(i-1)="":i=i-2:n=n-2                         [967]
2040 GOTO 1000                                      [339]
2050 '                                              [117]
2060 '***** L|schen *******                         [1255]
2070 '                                              [117]
2080 CLS#1:LOCATE #1,1,2:PRINT#1,CHR$(24)" [7001]
 Welche Nummer soll gel|scht werden "CHR$(
24);:INPUT #1," ";i
2090 IF i=0 OR i>n THEN CLS#1:PRINT CHR$(7 [5903]
):LOCATE#1,2,2:PRINT#1,"Falsche Eingabe !"
:FOR k=1 TO 2000:NEXT:GOTO 2080
2100 IF RIGHT$(a$(i),1)="*" THEN CLS#1:LOC [8327]
ATE #1,1,2:PRINT#1,i;". ";LEFT$(a$(i),LEN(
a$(i))-1);"     ";CHR$(24);"  ";CHR$(242);"
    richtig ?  < J/N >";CHR$(24);:GOTO 212
0
2110 CLS#1:LOCATE #1,1,2:PRINT#1,i;". ";a$ [6688]
(i);"     ";CHR$(24);"  ";CHR$(242);"    ri
chtig ?  < J/N >";CHR$(24);
2120 GOSUB 30                                       [852]
2130 IF M$="N" THEN 2080                            [837]
2140 IF LEN(A$(I))>35 THEN LOSCH=39 ELSE L [2987]
OSCH=LEN(A$(I))+5
2150 IF i<22 THEN sp=2:zei=i+1:ELSE sp=40: [2587]
zei=i-20
2160 '                                              [117]
2170 '***** L|schen auf Bildschirm *****            [3430]
2180 '                                              [117]
2190 LOCATE sp,zei:PRINT STRING$(losch,143 [3213]
);
2200 a$(i)=""                                       [229]
2210 CLS#1:LOCATE #1,1,2:PRINT#1,CHR$(24)" [4624]
 Noch einen Namen l|schen ?  < J/N > "CHR$
(24);:GOSUB 30
2220 IF (M$="N" AND h=1) THEN GOSUB 2270:G [2271]
OTO 1870
```

Einstellung
der
Druckparameter

(M)it / (O)hne Vornamen..... M
Kopien..................... 1
(S)chön- / (N)ormal-Druck... S

Änderung:Taste ↑ oder ↓ , dann <enter>
Copy → Drucken

```
2230 IF m$="N" THEN GOSUB 2270:GOSUB 2370: [2301]
GOTO 2480
2240 IF m$="J" THEN 2080                  [931]
2250 GOTO 2210                            [361]
2260 '                                    [117]
2270 '***** Durchschieben der Namen ***   [2329]
2280 '                                    [117]
2290 i=1                                  [423]
2300 WHILE i<>n+1                         [963]
2310 IF a$(i)="" THEN GOSUB 2350:i=i-1:n=n [1712]
-1
2320 i=i+1                                [444]
2330 WEND                                 [390]
2340 RETURN                               [555]
2350 FOR m=i TO n:a$(m)=a$(m+1):NEXT:RETUR [2624]
N
2360 '                                    [117]
2370 '***** Liste auf Schirm ******       [798]
2380 '                                    [117]
2390 MODE 2:INK 0,13:INK 1,0:PAPER 0:BORDE [3934]
R 13:WINDOW #1,1,80,24,25:PAPER#1,0:PEN#1,
1:CLS#1
2400 en=1:x1=0:y2=398:x2=638:y1=34:GOSUB 2 [2597]
700
2410 FOR i=1 TO n                         [456]
2420 IF i>21 THEN sp=40:zei=i-20:ELSE sp=2 [1937]
:zei=i+1
2430 IF RIGHT$(a$(i),1)="*" THEN LOCATE sp [6852]
,zei:PRINT CHR$(24)USING"##";i;:PRINT"."CH
R$(24)" ";LEFT$(a$(i),LEN(a$(i))-1):GOTO 2
450
2440 LOCATE sp,zei:PRINT CHR$(24)USING"##" [4132]
;i;:PRINT"."CHR$(24)" "LEFT$(a$(i),35)
2450 NEXT                                 [350]
2460 RETURN                               [555]
2470 '                                    [117]
2480 ' ******* Haupt-Menue ******         [1185]
2490 '                                    [117]
2500 CLS#1:LOCATE#1,3,2:PRINT#1,CHR$(7)CHR [9176]
$(24);" (A)bspeichern    (L)aden    (V)er{
nderung    (E)nde    (N)eu    (D)rucken ";
CHR$(24);
2510 GOSUB 30                             [852]
2520 IF M$="A" THEN 1660                  [583]
2530 IF m$="L" THEN GOSUB 70:GOTO 1500    [786]
2540 IF M$="V" THEN 2600                  [1293]
2550 IF M$="E" THEN CLS:END               [1338]
2560 IF M$="N" THEN CLEAR:GOSUB 70:GOTO 19 [2450]
0
2570 IF M$="D" THEN 2790                  [764]
2580 GOTO 2510                            [341]
2590 '                                    [117]
2600 '******** VER[NDERUNGS-MEN] *******  [1111]
2610 '                                    [117]
2620 H=0                                  [416]
2630 CLS#1:LOCATE#1,21,2:PRINT#1,CHR$(7)CH [5686]
R$(24)" (L)|schen     (H)inzuf]gen     (B)e
ides "CHR$(24);
2640 GOSUB 30                             [852]
2650 IF m$="L" THEN 2060                  [623]
2660 IF m$="H" THEN 1870                  [1157]
2670 IF m$="B" THEN H=1:GOTO 2060         [583]
2680 GOTO 2640                            [363]
2690 '                                    [117]
2700 '***** Umrahmung ******              [590]
2710 '                                    [117]
2720 FOR rr=0 TO en:PLOT x1+rr,y1,1       [1490]
2730 DRAW x1+rr,y2,1                      [663]
2740 DRAW x2-rr,y2,1                      [515]
2750 DRAW x2-rr,y1,1                      [503]
2760 DRAW x1+rr,y1,1:NEXT                 [1528]
2770 RETURN                               [555]
2780 '                                    [117]
2790 '******* Drucken ********            [940]
2800 '                                    [117]
2810 GOSUB 70:GOSUB 110                   [1376]
2820 CLS#1:LOCATE#1,15,1:PRINT#1,"Einstell [8648]
ung":LOCATE#1,19,2:PRINT#1,"der":LOCATE#1,
14,3:PRINT#1,"Druckparameter"
2830 GOSUB 3790                           [995]
2840 CLS:CLS#1:LOCATE#1,17,2:PRINT#1,"Druc [3201]
ken"
2850 druck=VAL(MID$(BIN$(INP(&F500),8),2,1 [2608]
))-1
2860 IF druck THEN 2890                   [815]
2870 CLS#3:LOCATE#3,7,1:PRINT#3,CHR$(7)CHR [7848]
$(24)" Bitte Drucker einschalten! ":LOCATE
#3,5,3:PRINT#3, "dann irgendeine Taste dr}
cken! "CHR$(24)
2880 CALL &BB06:CLS#3                     [608]
2890 CLS#2:GOSUB 110                      [676]
2900 anzahl=VAL (drckein$(2))             [1507]
2910 IF drckein$(1)<>"M" THEN ohne=1      [1836]
2920 IF anzahl<>1 THEN endlos=1:seite=1 EL [6246]
SE PRINT#8,CHR$(27);CHR$(56);:'*** Papierm
angel }bergehen *****
2930 IF drckein$(3)<>"S" THEN PRINT#8,CHR$ [7183]
(27);CHR$(120);CHR$(0);:ELSE PRINT#8,CHR$(
27);CHR$(120);CHR$(1);:'**** NLQ aus bzw.
ein
2940 PRINT#8,CHR$(27);CHR$(65);CHR$(10);CH [4246]
R$(27);CHR$(50);:'**** Zeilenabstand 10/72
 *****
2950 PRINT#8,CHR$(27);CHR$(67);CHR$(0);CHR [3527]
$(12);:'**** Formularh|he 12 ZOLL ******
2960                                      [0]
2970 PRINT#8,CHR$(27);CHR$(87);CHR$(1);:'* [5951]
****** Gedehnte Schrift ein ******
2980 IF freie=1 THEN uber$=ein$(2)        [1350]
2990 zent=(42-LEN(uber$))/2               [1114]
3000 CLS#2:zentbild=zent-3:LOCATE#2,zentbi [1497]
ld,2:PRINT#2,uber$
3010 PRINT#8:PRINT#8,TAB(ZENT);uber$:'**** [3494]
* }berschrift auf Drucker *******
3020 IF freie=1 THEN PRINT#8:GOTO 3070:'** [3498]
** erst bei Fach wieder ****
3030 ersatz$=LEFT$(x$(2),6)               [474]
3040 zent=(42-(LEN(ersatz$)+LEN(ein$(2))+1 [2207]
))/2
3050 zentbild=zent-3:LOCATE#2,zentbild,4:P [3399]
RINT#2,ersatz$;" ";ein$(2)
3060 PRINT#8,TAB(ZENT);ersatz$;" ";ein$(2) [3217]
:'**** Klasse bei Zentrierung auf Drucker
**
3070 PRINT#8,CHR$(27);CHR$(45);CHR$(1);:'* [3834]
**** Unterstreichen ein *****
3080 zent=(42-LEN(ein$(3)))/2             [1477]
3090 zentbild=zent-3:LOCATE#2,zentbild,6:P [2283]
RINT#2,ein$(3)
3100 PRINT#8:PRINT#8,TAB(ZENT);ein$(3)    [2270]
3110 PRINT#8,CHR$(27);CHR$(45);CHR$(0);:'* [4042]
**** Unterstreichen aus ******
3120 PRINT#8,CHR$(27);CHR$(87);CHR$(0);:'* [4239]
**** Gedehnt aus ******
3130 PRINT#8:PRINT#8                      [1314]
3140 PRINT#8,"      ";:PRINT#8,CHR$(27);CHR [4395]
$(61);:PRINT#8,STRING$(72,CHR$(26));:PRINT
#8,CHR$(0);
3150 FOR i=1 TO n                         [456]
3160 LOCATE#2,2,12:PRINT#2,CHR$(18);:PEN#2 [3004]
,3
3170 IF ohne=1 THEN q=INSTR(a$(i)," "):IF [9480]
 q<>0 THEN LOCATE#2,2,12:PRINT#2,CHR$(24);
USING"##";i;:PRINT#2,". ";LEFT$(a$(i),q);C
HR$(24):PEN#2,1:GOTO 3200
3180 IF RIGHT$(a$(i),1)="*" THEN LOCATE#2, [6135]
2,12:PRINT#2,CHR$(24);USING"##";i;:PRINT#2
,". ";LEFT$(a$(i),LEN(a$(i))-1);CHR$(24):P
EN#2,1:GOTO 3210
3190 LOCATE#2,2,12:PRINT#2,CHR$(24);USING" [5236]
##";i;:PRINT#2,". ";a$(i);CHR$(24):PEN#2,1
3200 IF ohne=1 AND q<>0 THEN PRINT#8,TAB(6 [4755]
);USING"##";i;:PRINT#8,". ";LEFT$(a$(i),q)
:q=0:GOTO 3230
3210 IF RIGHT$(a$(i),1)="*" THEN PRINT#8,T [5177]
AB(6);USING"##";i;:PRINT#8,". ";LEFT$(a$(i
),LEN(a$(i))-1):GOTO 3230
3220 PRINT#8,TAB(6);USING"##";i;:PRINT#8," [2904]
. ";a$(i)
3230 PRINT#8,"      ";:PRINT#8,CHR$(27);CHR [4395]
```

ANWENDUNG

```
$(61);:PRINT#8,STRING$(72,CHR$(26));:PRINT
#8,CHR$(0);
3240 IF i=35 THEN PRINT#8,CHR$(12);:´****   [4221]
ab 35 Namen Seitenvorschub *****
3250 NEXT                                    [350]
3260 IF endlos=1 THEN PRINT#8,CHR$(12);:EL  [3680]
SE GOTO 3280:´****** Seitenvorschub ******
3270 IF seite<anzahl THEN seite=seite+1:GO  [2680]
TO 2960:
3280 endlos=0:freie=0:seite=0:ohne=0:´****  [3758]
 Flags zur}cksetzen ******
3290 PRINT#8,CHR$(13);:PRINT#8,CHR$(27);CH  [4888]
R$(120);CHR$(1);:´**** NLQ einschalten ***
***
3320 ´                                       [117]
3330 ´***** Men] f]r Druck-Wiederholung **  [2135]
***
3340 ´                                       [117]
3350 GOSUB 70:GOSUB 110                     [1376]
3360 CLS#2:LOCATE #2,11,7:PEN#2,3:PRINT#2,  [15653]
"N";:PEN#2,1:PRINT#2,"eu-Druck..":LOCATE #
2,11,9:PEN#2,3:PRINT#2,"E";:PEN#2,1:PRINT#
2,"nde.......":LOCATE #2,11,11:PEN#2,3:PRI
NT#2,"H";:PEN#2,1:PRINT#2,"auptmen]..":GOS
UB 150
3370 GOSUB 30                                [852]
3380 IF m$="N" THEN GOSUB 3430:GOTO 2790    [1988]
3390 IF m$="E" THEN CLS:END                 [1338]
3400 IF m$="H" THEN GOSUB 2370:GOTO 2480    [1476]
3410 GOTO 3370                               [455]
3420 ´                                       [117]
3430 ´******* Neueingabe Druck-]berschrift  [2336]
en ********
3440 ´                                       [117]
3450 RESTORE 250                             [785]
3460 FOR m=1 TO 5:READ x$(m):NEXT           [1421]
3465 ´m$="F":GOTO 3510                       [592]
3470 GOSUB 1410:´***** Abfrage welche }ber  [4248]
schrift ******
3480 GOSUB 30                                [852]
3490 IF m$="K" THEN uber$="Klassenliste":G  [3908]
OTO 3560
3500 IF m$="P" THEN uber$="Klassenpflegsch  [4239]
aft":x$(3)=x$(4):GOTO 3560
3510 IF m$="F" THEN x$(2)=x$(5):x$(3)=x$(4  [3102]
):freie=1:GOTO 3560
3520 GOTO 3480                               [481]
3530 ´                                       [117]
3540 ´******* Men} Druck-]berschriften (nd  [3106]
ern ********
3550 ´                                       [117]
3560 CLS#2                                   [372]
3570 z=7:FOR m=1 TO 3:LOCATE#2,15,z:z=z+2:  [4920]
PRINT#2,ein$(m):NEXT
3580 rev=1:losch=7                           [855]
3590 GOSUB 4050                             [1015]
3600 GOSUB 3750                             [1035]
3610 a=240                                   [592]
3620 WHILE a<>224:´*** Copy ***             [1553]
3630 a$=INKEY$:IF a$="" THEN 3630           [1443]
3640 a=ASC(a$)                              [1036]
3650 IF a=13 THEN GOSUB 3710:´***Enter***   [2442]
3660 IF a=241 THEN rev=MIN(rev+1,3):losch=  [3800]
MIN(losch+2,11):´** runter **
3670 IF a=240 THEN rev=MAX(rev-1,1):losch=  [3982]
MAX(losch-2,7):´*** rauf **
3680 GOSUB 3750                             [1035]
3690 WEND                                    [390]
3700 RETURN                                  [555]
3710 GOSUB 4080:LOCATE#2,15,losch:PRINT#2,  [6120]
CHR$(18);:LOCATE#2,15,losch:CLEAR INPUT:LI
NE INPUT#2,"",ein$(rev)
3720 IF ein$(rev)="" OR LEN(ein$(rev))=1 T  [1894]
HEN 3740
3730 m$=ein$(rev):GOSUB 890:ein$(rev)=m$    [2210]
3740 GOSUB 4050:RETURN                      [1309]
3750 z=7:FOR m=1 TO 3                       [1383]
3760 IF m=rev THEN LOCATE#2,1,z:PRINT#2,CH  [6063]
R$(24);" ";x$(m);CHR$(24);:z=z+2:ELSE LOCA
TE#2,1,z:z=z+2:PRINT#2," ";x$(m);
3770 NEXT:RETURN                             [940]
3780 ´                                       [117]
3790 ´******* Druck-Parameter ********      [1014]
3800 ´                                       [117]
3810 drck$(1)="(M)it / (O)hne Vornamen....  [9752]
.":drck$(2)="Kopien.....................":
:drck$(3)="(S)ch|n- / (N)ormal-Druck...":d
rckein$(1)="M":drckein$(2)="1":drckein$(3)
="S"
3820 CLS#2                                   [372]
3830 z=7:FOR m=1 TO 3:LOCATE#2,33,z:z=z+2:  [4693]
PEN#2,3:PRINT#2,drckein$(m):PEN#2,1:NEXT
3840 rev=1:losch=7                           [855]
3850 GOSUB 4050                             [1015]
3860 GOSUB 3990                              [987]
3870 a=240                                   [592]
3880 WHILE a<>224                           [1106]
3890 a$=INKEY$:IF a$="" THEN 3890           [1534]
3900 a=ASC(a$)                              [1036]
3910 IF a=13 THEN GOSUB 3970                [1321]
3920 IF a=241 THEN rev=MIN(rev+1,3):losch=  [3255]
MIN(losch+2,11)
3930 IF a=240 THEN rev=MAX(rev-1,1):losch=  [3370]
MAX(losch-2,7)
3940 GOSUB 3990                              [987]
3950 WEND                                    [390]
3960 RETURN                                  [555]
3970 GOSUB 4080:LOCATE#2,33,losch:PRINT#2,  [7311]
CHR$(18);:LOCATE#2,33,losch:CLEAR INPUT:IN
PUT#2,"",drckein$(rev):drckein$(rev)=UPPER
$(drckein$(rev))
3980 GOSUB 4060:GOSUB 4050:RETURN           [2363]
3990 z=7:FOR m=1 TO 3                       [1383]
4000 IF m=rev THEN LOCATE#2,2,z:PRINT#2,CH  [5760]
R$(24);" ";drck$(m);CHR$(24);:z=z+2:ELSE L
OCATE#2,2,z:z=z+2:PRINT#2," ";drck$(m);
4010 NEXT:RETURN                             [940]
4020 ´                                       [117]
4030 ´***************** Meldungen *********  [2059]
***************
4040 ´                                       [117]
4050 CLS#3:LOCATE #3,2,1:PEN#3,1:PRINT#3,"  [12375]
[nderung:";:PEN#3,2:PRINT#3,"Taste ";CHR$(
240);" oder ";CHR$(241);" , dann <enter>":
LOCATE#3,14,3:PEN#3,1:PRINT#3,"Copy ";:PE
N#3,2:PRINT#3,CHR$(243);" Drucken";:RETUR
N
4060 IF drckein$(2)<>"1" OR n>35 THEN PEN#  [9233]
2,3:LOCATE#2,9,14:PRINT#2,CHR$(7)"Endlospa
pier einlegen";:PEN#2,1:ELSE LOCATE#2,9,14
:PRINT#2,CHR$(18);
4070 IF drckein$(1)="0" THEN drckein$(1)="  [3090]
O":RETURN
4080 CLS#3:LOCATE#3,4,2:PRINT#3,"Die Einga  [5915]
be mit <enter> abschlie~en":RETURN
4090 ´                                       [117]
4100 ´************** Ausgabe Progr. Name **  [2033]
************
4110 ´                                       [117]
4120 INK 1,24:INK 2,20:INK 3,6:INK 4,0:BOR  [2099]
DER 1
4130 MODE 0:WINDOW#1,3,18,2,4:WINDOW#2,2,1  [3330]
9,11,15:PAPER#1,4:PAPER#2,2
4140 CLS#1:PEN#1,3:LOCATE#1,6,2:PRINT#1,"L  [3223]
ISTE";
4150 CLS#2:PEN#2,4:LOCATE#2,7,2:PRINT#2,CH  [5450]
R$(164);" 1985":PRINT#2," Wolfgang Tewes"
;
4160 FOR k=1 TO 1500:NEXT                   [1793]
4170 RETURN                                  [555]
4180 ´                                       [117]
4190 ´********* UMLAUTE *******             [1489]
4200 ´                                       [117]
4210 SYMBOL AFTER 90                        [1214]
4220 SYMBOL 123,108,0,120,12,124,204,118,0  [2131]
4230 SYMBOL 91,102,24,60,102,126,102,102,0  [2353]
4240 SYMBOL 124,102,0,60,102,102,102,60,0   [2072]
4250 SYMBOL 92,102,56,108,198,198,108,56,0  [2215]
4260 SYMBOL 125,102,0,102,102,102,102,62,0  [2418]
4270 SYMBOL 93,102,0,102,102,102,102,60,0   [2333]
4280 SYMBOL 126,120,198,198,252,198,198,24  [2253]
8,192
4290 KEY DEF 17,1,123,123                    [975]
4300 KEY DEF 19,1,124,124                   [1225]
4310 KEY DEF 22,1,125,125                   [1232]
4320 KEY DEF 26,1,126,124                    [691]
4330 RETURN                                  [555]
4340 ´                                       [117]
4350 ´********** Fehler-Routine **********  [2363]
*
4360 ´                                       [117]
4370 IF DERR=146 THEN PRINT CHR$(7)CHR$(24  [5560]
)" Falscher Datei-Name! "CHR$(24):RESUME 1
550
4380 MODE 2:INK 0,13:INK 1,0:PAPER 0:BORDE  [3934]
R 13:WINDOW #1,1,80,24,25:PAPER#1,0:PEN#1,
1:CLS#1
4390 RESUME 2500:´******* zum Hauptmen}     [2624]
```

Interrupts auf dem CPC

Wer sich schon einmal gefragt hat, wie es in Spielprogrammen eigentlich gelöst wird, daß neben dem laufenden Programm noch eine Melodie ertönt, oder woher der Rechner weiß, ob in einem laufenden Programm die ESC-Taste gedrückt wurde, der wird irgendwann einmal auf den Begriff INTERRUPT stoßen. Ein Interrupt ist den im Rechner periodisch auftretendes Ereignis, welches zu Deutsch Unterbrechung heißt und dazu verwendet wird, Unterprogramme anzuspringen.

Die Programmierung eines Interrupts erfolgt über den Interrupt-Handler und kann beim CPC sehr einfach durch die Verwendung von Event-Blöcken vorgenommen werden. Auch unter Basic finden solche Ereignisse durch die Befehle EVERY, DI, EI und ON SQR Verwendung
Da sich das Handbuch über die Verwendung dieser Befehle sehr ausführlich ausläßt, wollen wir uns nur mit der Programmierung unter M-Code befassen.
Die Interrupt-Steuerung wird über Events und Chains vorgenommen. Zum eigentlichen Auslösen eines Interrupts ist jedoch zunächst einmal ein Signal der Hardware erforderlich.

Hardware Grundlagen

Ein Interrupt ist, wie bereits eingangs erwähnt, ein Aufruf eines Unterprogramms. Im Gegensatz zu normalen Programmaufrufen startet man einen Interrupt jedoch nicht mit einem CALL, sondern läßt das von der CPU automatisch erledigen. Der Z80A hat nämlich unter seinen vielen Beinchen auch eines, das die Bezeichnung IRQ (Interrupt Request) trägt.
An diesem Beinchen wird nun von einem Timer, der sogenannten Interrupt-Quelle, von Zeit zu Zeit ein Signal angelegt, das der CPU mitteilt, daß nun ein Interrupt freigegeben werden kann. Ist dieses Signal LOW (0 Volt), so arbeitet der Prozessor den letzten Befehl ab,und sieht danach an Speicherstelle &0038 nach, was für einen Interrupt er abarbeiten soll.
Im Gegensatz zu anderen Z80-Systemen kennt der Schneider nur eine solche Inerrupt-Quelle, das Gate Array.
Deshalb arbeitet der CPC auch nur im Interrupt-Modus 1 (IM1) und bearbeitet bei Auslösung auf jeden Fall zunächst einmal die Speicherstelle &0038. Dort kann dann natürlich ein CALL stehen, der den Interrupt weiterleitet.
Der eigentliche Softwareteil der Interruptbehandlung geschieht im KERNEL, in der Firmware und bei der von Ihnen geschriebenen Interruptroutine im RAM.

Software Grundlagen

Nachdem der Interrupt hardwaremäßig ausgelöst wurde und die Adresse &0038 erreicht hat, wechselt er zunächst einmal seinen Namen und heißt zum besseren Verständnis nunmehr EVENT (Ereigniss). Dabei unterscheidet man zunächst einmal zwei Sorten von Events.

Der synchrone Event

Ein synchroner Event steht im Kernel in einer verketteten Liste, dem SPQ (Synchron Pending Queue).
Alle diese Events sind nach festgelegten Prioritäten geordnet und haben keine andere Aufgabe, als zunächst einmal Wegweiser zum eigentlich abzuarbeitenden Event zu sein. Da diese Events zu Blöcken mit Prioritätsmarkierungen und Koppeladressen zusammengefaßt sind und es deren mehrere geben kann, wird von einem Queue gesprochen.
Ein Queue wird im Billard verwendet, und ist ein langer Stab, mit dem man Kugeln anschieben kann. Da auch die Interrupts "angeschoben" (man spricht vom Ankicken) werden müssen, nahm man sinnigerweise diesen Namen. Weiterhin besteht der Begriff SPQ noch aus dem Wort Pending, was soviel wie schwebend heißt. Tatsächlich "schwebt" die Abarbeitung dieser Interrupt-Liste so lange, bis vom Kontrollprogramm (Basic Interpreter usw.) die Erlaubnis zur Ausführung kommt.
Die Events laufen also synchron zum Kontrollprogramm.
Typisch für einen solchen synchronen Event ist beispielsweise der Break-Event, der auf den Keyboard Manager im Kernel losgelassen wird und abfragt, ob ESC gedrückt wurde.
Dieser Interrupt-Block wird zwar laufend angesprungen, jedoch erst abgearbeitet, wenn der Interpreter mit der Abarbeitung seiner momentan laufenden Schleife fertig ist.
Das heißt, daß die ESC-Taste nur zwischen zwei Basic-Befehlen zum Einsatz kommt und Sie nicht mitten in einer Sprunganweisung abbrechen können.

Asynchrone Events

Ein asynchroner Event ist zunächst einmal genau das gleiche wie ein synchroner Event. Auch diese Events sind in einer Liste miteinander verkettet. Allerdings heißt diese Liste nicht mehr SPQ, sondern APQ, was logischerweise Asynchronus Pending Queue bedeutet.
Asynchrone Events sind nicht kontrollierbar. Sie werden zwar auch der Reihe nach abgearbeitet, lassen sich jedoch vom Kontrollprogramm nicht stören. Diese Events, die ungefähr 300 mal pro Sekunde ausgelöst werden, bieten die Möglichkeit der Echtzeitprogrammierung, können aber auch dazu führen, daß sie bei ungeschickter Programmierung nie wieder zum Kontrollprogramm zurückkehren.
Dummerweise wird ein Event, der in einem PQ untergebracht ist, nach seiner Abarbeitung aus demselben entfernt.
Das heißt im Klartext, daß die periodische Abarbeitung eines asynchronen Events ohne ständiges Erneuern des PQ nicht möglich ist.
Um ein PQ im Takt von 300 Hz jedoch nach jeder Abarbeitung neu zu installieren, benötigt man wiederum einen Interrupt, der sich wiederum löscht.
Um nun dennoch einen periodische PQ aufzubauen, verfügt der CPC über die sinnvolle Einrichtung von CHAINS.

CHAINS und KETTEN

Chains sind eine weitere Listenform, in der der Programmierer alle gewünschten PQ-Blöcke ablegen kann. Eine solche Chain wird bei jedem Interrupt durchgegangen und die entsprechenden Blöcke in die PQ eingehangen, von wo aus sie angekickt werden können.
Genau genommen hat der CPC gleich drei solcher Chains, die ähnlich wie die Events, unterschiedliche Prioritäten haben. Da ist zum ersten

die "Fast Ticker Chain" (FTC). Wie der Namensteil FAST schon vermuten läßt, ist dies der schnellste Event, den wir auslösen können. Tatsächlich kickt die FTC alle 300-stel Sekunde einen Block in den PQ.
Etwas langsamer ist dagegen schon die normale Ticker Chain (TC). Sie kickt mit einer Frequenz von 50 Hz, die der deutschen Netzspannungsfrequenz entspricht. Die Ticker Chain zählt dazu immer sechs Interrupts hoch und löst dann aus.
Außerdem gibt es noch die "Frame Fly Chain", die immer dann kickt, wenn der Bildschirm gerade einen VSYNC-Impuls abgeschlossen hat und der Elektronenstrahl des Monitors von der linken unteren Bildschirmecke in die obere rechte zurückläuft. Mit dieser FFC ist es möglich, synchrone Events zu starten, ohne daß der Monitor anfängt zu flimmern. Eine solche Chain ist ähnlich aufgebaut wie ein PQ. Sie besteht aus den Interrupt-Blöcken, die ihre Prioritätsmerkmale, ihre Destinationsadresse sowie eine Koppeladresse, die auf den nächsten Chain-Block hinweist, enthält.
Die Blöcke für die TC enthalten noch einige Zähler. Der Header (Kopf) eines solchen Event-Blocks hat einen grundsätzlichen Aufbau, der eigentlich auch schon das ganze Geheimnis der Interrupt-Programmierung ausmacht.

Der Interrupt Block Header

Ein Block-Header hat grundsätzlich folgenden Aufbau:
BYTE 0 - 1
Pointer auf nächsten Interrupt-Block
BYTE 2
Counter für den PQ
BYTE 3
Priorität des Blocks
BYTE 6
ROM-Konfiguration (Optional)
BYTE 7–
Benutzerfeld

Die Bits des Priorität-Bytes haben dabei folgende Bedeutung:

Bit	Zust.	Auswirkung
0	0	Sprung nach einer NEAR ADRESS
0	1	Sprung nach einer FAR ADRESS
1 - 4	0/1	Priorität für synchrone Events
5	0	SYNC EVENT ON
5	1	SYNC EVENT OFF
6	0	Standard-Event
6	1	Express-Event
7	0	synchroner Event
7	1	Asynchroner Event

Wie eingangs erwähnt, verzweigt der Pointer, also die ersten beiden Bytes des Block-Headers, auf den nächsten Block, für den wieder die gleichen Daten zutreffen.
Diese Bytes errechnet Ihr CPC netterweise automatisch und Sie sollten sich hüten, dort irgendwelche Angaben zu machen, da sie Ihren Interrupt nach Abarbeitung ganz schön "in die Wüste schicken" können. Wichtig ist nur noch das siebente Bit des Pointers. Ist dieses nämlich gesetzt (1), so ist dieser Event ausgeschaltet. Ist das Bit 0, so übernimmt der PQ-Counter die Verwaltung und gibt an, wie oft dieser Event ausgeführt werden soll. Zumindest hatten sich die Programmierer von Locomotive das so gedacht! Durch einen ROM-Fehler klappt das allerdings nicht so ganz. Der Event wird überhaupt nur dann in den PQ übernommen, wenn der PQ-Counter leer ist. Sobald irgend etwas anderes darin steht, ist der Interrupt gesperrt.
Das ist zwar schade, aber man kann den PQ-Counter ja immer noch als Flag benutzen.

Eine "Near-Address" sollte jedem Assemblerprogrammiere ein Begriff sein. Sie gibt an, ob die Interruptroutine im RAM von &4000 bis &BFFF steht, oder bei Angabe einer Far-Address in ein UPPER ROM gesprungen werden soll.
Wurde ein FAR CALL ausgewählt, so muß in BYTE 6 noch eine ROM-Konfiguration nach den für FAR CALLs gültigen Bedingungen übergeben werden. Ab BYTE 6 steht dann das Benutzerfeld. Dieses Feld kann beliebig lang sein, die Entfernung zum nächsten Block errechnet der CPC ja von alleine.
Dieses Benutzerfeld entspricht der eigentlichen Interruptroutine, die ausgeführt werden soll.
Hier kann ein CALL, ein JUMP oder eine ganze Routine stehen. Der PQ arbeitet hier auf jeden Fall weiter.
Eine Besonderheit des Benutzerfeldes ist noch, daß bei einem NEAR CALL vom Kernel ein Pointer nach DE geschrieben wird, der auf die beiden Bytes vor dem Benutzerfeld hinweist. Hier können für die Interruptroutine noch Parameter abgelegt werden. Bei einem FAR CALL steht dieser Pointer in HL und geht sogar drei Bytes vor das Benutzerfeld.
Wie Sie sehen, geht das Einhängen von Interrupts auf sehr einfache Weise. Lediglich von den ersten beiden Bytes für die Ankopplungsadresse des nächsten Blocks sollte man die Finger lassen. Die oben genannten Daten gelten jedoch nur für die FTC und die FFC. Beim TC verhält sich das alles noch ein klein wenig anders.

Ticker Chains

Der Header eines TC-Blocks hat folgenden Aufbau:
BYTE 0/1
Pointer auf nächsten Block
BYTE 2/3
TICKER COUNTER
BYTE 4/5
RELOAD COUNTER
BYTE 6–
siehe Block-Header

Byte 2 und 3 enthalten einen 16-Bit-Wert, der bei jedem auftretenden Ticker um 1 vermindert wird. Das heißt, daß alle 50-stel Sekunde 1 abgezogen wird. Ist der Counter leer, wird gekickt. Steht also im Ticker-Counter eine 50, so müssen 50 Ticker ablaufen (entspricht 50/50 also eine Sekunde), sodann wird der Interrupt ausgelöst. So ließe sich zum Beispiel auf einfachste Weise eine Echtzeituhr realisieren. Damit der Zähler nach Ablauf der Ticker-Zeit auch wieder einen neuen Startwert erhält, kann in den Bytes 4 und 5 ein sogenannter Reload-Wert eingesetzt werden, der die Counter Bytes wieder füllt.

How to do it

Um nun diese Interrupts auch so komfortabel wie möglich in die Chains und PQ's einzubinden, hält das Betriebssystem noch eine Reihe von Firmware-Routinen bereit, über deren Handhabung Sie etwas in der Referenzliste 1 erfahren.

Interrupts Off!! Der Autor meldet sich zu Wort

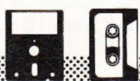
Das soeben Gelesene macht auf den ersten Blick einen sinnverwirrenden Eindruck. Trotzdem sollten Sie sich von den Interrupts nicht abschrecken lassen. Bei näherer Betrachtung kommt nämlich Struktur in das Ganze, und so mancher Besitzer eines anderen Z80-Systems neidet dem CPC bestimmt seine komfortablen Einrichtungen zur Behandlung dieser

doch sehr zeitkritischen Angelegenheit. Befassen Sie sich zunächst einmal mit der Behandlung von Interrupts in Basic (Absturzsicher) und Sie werden bald Parallelen zur Interruptbehandlung in Maschinensprache finden. Als weiterführende Literatur sei das Firmwarehandbuch der Firma Schneider empfohlen.
Viel Spaß und Erfolg beim Programmieren wünscht

(TM)

Adresse	Syntax	Auswirkung
&BCD7	Einsprung: HL=Adresse Bildblock, B=Eventklasse, C=ROM-Select, DE=Eventroutine. Aussprung: AF, DE, HL zerstört.	Initialisierung und Übergabe eines Bildblocks an die Bildaufbauliste.
&BCDA	Einsprung: HL=Bildaufbaublock. Aussprung: AF, DE, HL zerstört.	Übergabe eines Bildblocks an die Bildaufbauliste ohne Initialisierung.
&BCDD	Einsprung: HL=Bildblock. Aussprung: AF, DE, HL zerstört.	Löschen eines Bildblocks in der Bildaufbauliste.
&BCE0	Einsprung: HL=Fast-Ticker-Block, B=Eventklasse, C=ROM-Select, DE=Eventroutine. Aussprung: AF, DE, HL zerstört.	Initialisierung und Übergabe eines Blocks an die Fast-Ticker-Liste.
&BCE3	Einsprung: HL=Fast-Ticker-Block. Aussprung: AF, DE, HL zerstört.	Übergabe eines Blocks an die Fast-Ticker-Liste ohne Initialisierung.
&BCE6	Einsprung: HL=Fast-Ticker-Block. Aussprung: AF, DE, HL zerstört.	Löschen eines Blocks in der Fast-Ticker-Liste.
&BCE9	Einsprung: HL=Ticker-Block, DE=Start Counter Value BL=Repeat Value. Aussprung: AF, BC, DE zerstört.	Block an die normale Ticker-Liste übergeben.
&BCEC	Einsprung: HL=Ticker-Block. Aussprung: Block gefunden: carry on, DE=Event Counter State. Block nicht gefunden: carry off, DE zerstört. Immer: A, HL, Flags zerstört.	Löschen eines Blocks in der normalen Ticker-Liste.
&BCEF	Einsprung: HL=Eventblock, B=Eventklasse, IC=ROM-Select, DE=Eventroutine. Aussprung: HL enthält Adresse des Eventblocks plus einen Offset von +7.	Initialisiere einen Eventblock.
&BCF2	Einsprung: HL=Eventblock. Aussprung: AF, BC, DE, HL zerstört.	Starten eines Eventblocks (kann über Low Jump vom Interrupt-Pfad aus gestartet werden).
&BCF5	Einsprung: Keine Bedingungen. Aussprung: AF, HL zerstört.	Lösche die Warteliste für synchrone Events.
&BCF8	Einsprung: HL = Eventblock. Aussprung: AF, BC, DE, HL zerstört.	Lösche ein synchrones Ereignis aus der Warteschlange für synchrone Ereignisse.
&BCFB	Einsprung: keine. Aussprung: Event gefunden: Carry on, HL = Eventblock, A = Priorität. Kein Event: Carry off, A, HL zerstört. Immer: DE zerstört.	Hole den nächsten Event aus der Warteschlange.
&BCFE	Einsprung: HL = Eventblock. Aussprung: AF, BC, DE, HL zerstört.	Bearbeite Event.
&BD01	Einsprung: C = Priorität, HL = Eventblock. Aussprung: AF, BC, DE, HL zerstört.	Stoppe die Bearbeitung eines Events.
&BD04	Einsprung: keine. Aussprung: HL zerstört.	Sperren von synchronen Events.
&BD07	Einsprung: keine. Aussprung: HL zerstört.	Freischalten der synchronen Events.
&BD0A	Einsprung: HL = Eventblock. Aussprung: AF zerstört.	Sperre einen bestimmten Event.
&BD0D	Einsprung: keine. Aussprung: DE, HL = 4 Byte Counter.	Gib die Zeit seit Einschalten des Rechners in 1/300-stel Sekunden zurück.
&BD10	Einsprung: keine. Aussprung: AF zerstört.	Setzen des Time-Counters.

Tape Monitor



Mit dem Programm Tape Monitor wird es dem User ermöglicht, einzelne Blöcke eines Programms zu laden, zu verändern und wieder abzuspeichern. Der Tape Monitor ermöglicht es auch, Programme mit Ladefehlern ohne großen Datenverlust wieder herzustellen, indem ein fehlerhafter Block bis zu seinem Ladefehler korrekt geladen wird und die fehlenden Daten durch Nullbytes aufgefüllt werden.
Der Tape Monitor ist sehr einfach zu bedienen. Im Window #2, in der rechten oberen Bildschirmecke, steht permanent das Menue bzw. der derzeit ausgewählte Menuepunkt. Wurde ein Programm geladen, zeigt das Window #1 immer die aus dem Header errechneten Programmdaten, wie Startadresse, Länge, Autostart usw.
Ein Programmblock wird in acht Seiten zu je 256 Bytes unterteilt, die im Menuepunkt Ändern-Programm jederzeit direkt über die Tasten 0 - 7 oder über die Pfeiltasten aufgerufen werden können. Vor den ausgelisteten Zeilen steht jeweils die Adresse, ab der die Seite des derzeit geladenen Blocks steht, wenn das Programm regulär geladen wird. Beim Header wird einfach von Null an gezählt. Im Editor kann man wahlweise die einzelnen Bytes hexadezimal verändern (wobei stets beide Nibble neu geschrieben werden müssen), oder die Bytes direkt im ASCII-Feld verändern. Der Cursor wird über die Pfeiltasten gesteuert. Über die TAB-Taste wird der Editor wieder verlassen, und die Seite des aktuellen Blocks noch einmal ausgelistet.
Alle momentan möglichen Befehle werden im Window #1 angezeigt. Zum Abspeichern stehen dem User, außer den normalen Geschwindigkeiten 0 und 1, noch eine dritte mit etwa 3000 Baud zur Verfügung.
Der Header wird ab der Adresse &4E00, das Programm ab &5000 abgelegt. Dadurch wird es ermöglicht, ein zusätzliches Programm (z.B. Arnor A/D/E) zu laden.
Der Tape Monitor ist überwiegend in Basic geschrieben. Die eingesetzten Maschinenprogramme dienen zum Speicherbereich löschen, Programm laden/sichern und zum Setzen der Bandgeschwindigkeit.

(A. Stolzenberger)

Variablenliste:

i	Universelle Laufvariable
me$ (1-3)	Hauptmenuepunkte
mu$ (1-3)	Untermenuepunkte
a$	Universalvariable für Tastaturabfrage
a	ASCII-Wert von a$
t	Zweite universelle Laufvariable, falls i besetzt ist
b	Enthält den Wert der aktuellen Seite eines Blocks. Trägt Werte von 0 - 7
he	Enthält den Wert &4e00 ab dem der Header im Speicher steht
hd	Header Flag; Information für Dump-Unterprogramm, ob der Header oder das Programm gelistet wird (Header=-1 Programm=0)
s	Enthält bei Programmlisting den Wert, ab welchem die Seite des Blocks normalerweise im Speicher stünde (Header: s=0)
b	Enthält den Wert, ab dem die Seite des Programmblocks oder der Header tatsächlich im Speicher steht
y	Zeile des Cursors
x	Spalte des Cursors
xx	Tatsächliche Spalte des Cursors, errechnet sich aus x
fl	Flag; enthält den Wert -1, wenn der obere Nibble eines Bytes beschrieben wurde
r	Enthält den von der Tastatur übernommenen Wert, der in den Speicher geschrieben wird
rr	Hilfsvariable für r, übernimmt den oberen Nibble-Wert, bis der untere eingegeben wurde und in r vorliegt

Unterprogramme:

110 - 230	Schreibt das Maschinenprogramm in den Speicher
240 - 360	Setzt Mode, Farbe und baut den Bildschirm auf
370 - 500	Hauptmenue, verzweigt in die Routinen für Ändern, Laden und Speichern
540 - 590	Ruft das Maschinen UP zum Lesen eines Blockes auf
630 - 730	Wertet die Header-Information aus und zeigt diese im Window 1
770 - 880	Setzt die Bandgeschwindigkeit und ruft die Maschinenroutine zum Speichern auf
920 - 1050	Untermenue, verzweigt zum Programm bzw. Header-Dump und Editor oder zurück zum Menue
1130 - 1160	Listet den Header und verzweigt zum Editor oder zum Untermenue
1200 - 1310	Listet die erste Seite des Programmblocks, verzweigt zum Editor oder zum Untermenue, oder wählt eine neue Seite eines Blocks an
1350 - 1430	Listet eine Programmseite oder den Header
1470 - 1610	Editor für Programm oder Header
1650 - 1710	Berechnet tatsächliche Cursorspalte
1750 - 1800	Liest einen Wert zwischen 0 und F von der Tastatur
1840 - 1950	Liest den ASCII-Wert einer Taste von der Tastatur

&BC6E	CAS START MOTOR
&BCA1	CAS READ
&BC9E	CAS WRITE
&BC68	CAS SET SPEED
&BB81	TXT CUR ON

```
10 '********************************       [1285]
20 '*    T A P E - M O N I T O R   *       [329]
30 '*     by Andi Stolzenberger    *       [2212]
40 '*        (c) Februar 1986      *       [1174]
50 '********************************       [1285]
60 '                                       [117]
70 '                                       [117]
80 '                                       [117]
90 '                                       [117]
100 '********************************      [1285]
110 '*         Mpg einlesen         *      [1299]
120 '********************************      [1285]
130 '                                      [117]
140 '                                      [117]
150 MEMORY &4CFF                           [114]
160 FOR i=&4D00 TO &4D00+52:READ a$:POKE i [4176]
,VAL("&"+a$):NEXT
170 DATA cd,6e,bc,21,00,4e,11,ff,00,3e,2c, [3710]
```

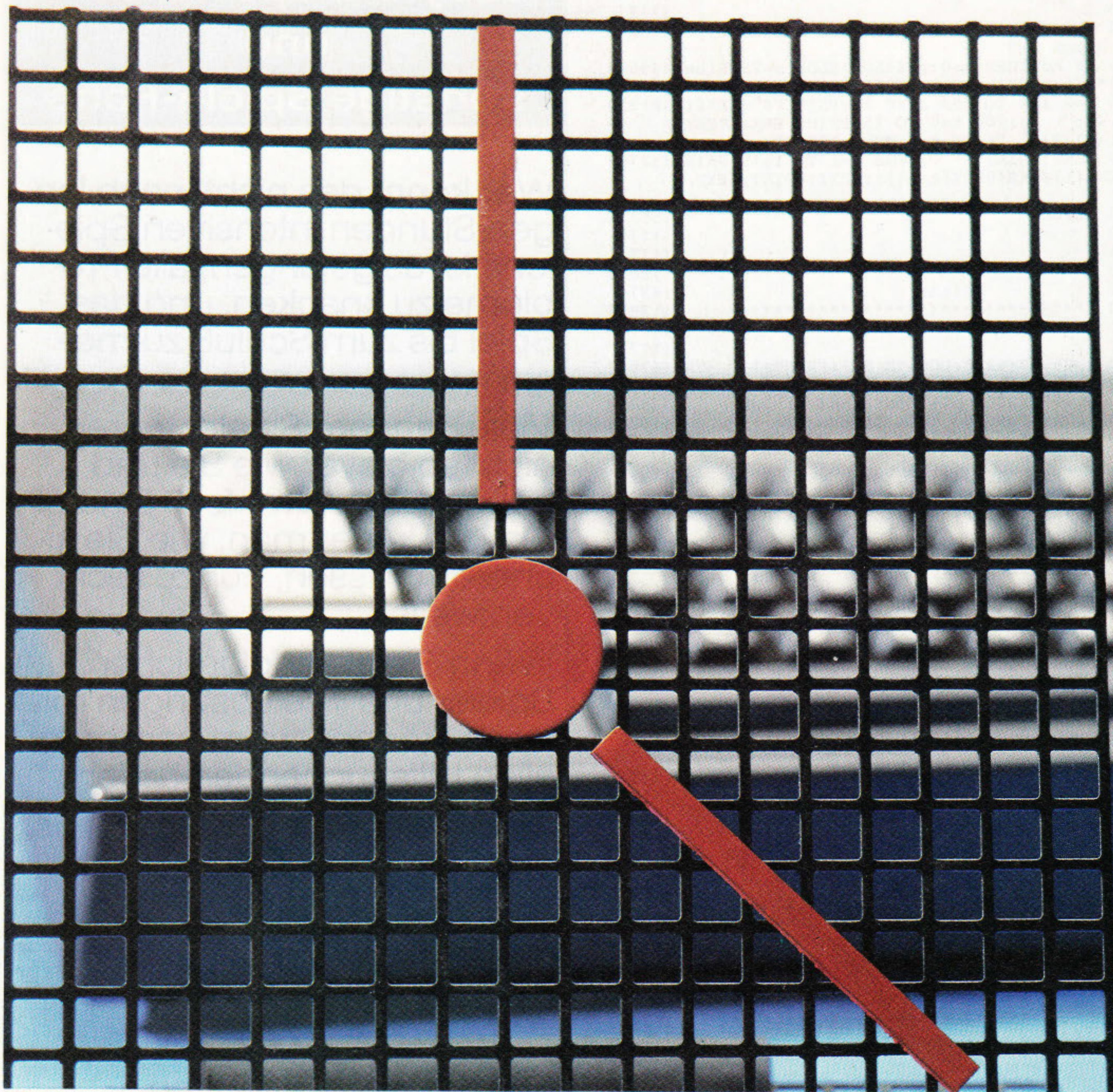
```
cd,a1,bc,21,00,50,ed,5b,13,4e
180 DATA 3e,16,cd,a1,bc,cd,71,bc,c9            [1969]
190 DATA 21,00,50,11,01,50,01,00,08,36,00,     [4266]
ed,b0,c9,21,82,00,3e,01,cd,68,bc,c9
200 me$(1)="<1>  Laden":me$(2)="<2>  Spe       [10979]
ichern":me$(3)="<3>  Aendern":mu$(1)="<1>
 Header":mu$(2)="<2>  Programm":mu$(3)=
"<3>  Menue"
210 '                                          [117]
220 '                                          [117]
230 '********************************          [1285]
240 '*      Bildschirm aufbauen      *         [1920]
250 '********************************          [1285]
260 '                                          [117]
270 '                                          [117]
280 MODE 2:INK 0,13:INK 1,0:BORDER 13:CALL     [2503]
 &4D2C
290 PRINT CHR$(150)STRING$(78,154)CHR$(156     [2155]
);
300 FOR i=2 TO 6:LOCATE 1,i:PRINT CHR$(149     [4350]
);:LOCATE 80,i:PRINT CHR$(149);:NEXT
310 LOCATE 1,7:PRINT CHR$(151)STRING$(78,1     [11743]
54)CHR$(157);:LOCATE 58,1:PRINT CHR$(158);
:LOCATE 58,7:PRINT CHR$(155);:FOR i=2 TO 6
:LOCATE 58,i:PRINT CHR$(149);:NEXT
320 FOR i=8 TO 24:LOCATE 1,i:PRINT CHR$(14     [7093]
9);:LOCATE 80,i:PRINT CHR$(149);:NEXT:LOCA
TE 1,25:PRINT CHR$(147)STRING$(78,154)CHR$
(153);
330 WINDOW 2,79,8,24:WINDOW#1,2,57,2,6:WIN     [2774]
DOW#2,59,79,2,6
340 '                                          [117]
350 '                                          [117]
360 '********************************          [1285]
370 '*          Hauptmenue           *         [1375]
380 '********************************          [1285]
390 '                                          [117]
400 '                                          [117]
410 CLS#2:PRINT#2,"    Menue":PRINT#2:FOR      [6111]
i=1 TO 3:LOCATE#2,2,i+2:PRINT#2,me$(i):NEX
T
420 a$="":WHILE a$="":a$=INKEY$:WEND           [2835]
430 IF a$<"1" OR a$>"3" THEN 420               [2607]
440 i=VAL(a$):LOCATE#2,2,2+i:PRINT#2,CHR$(     [3076]
24)me$(i)CHR$(24)
450 LOCATE #2,12,1:PRINT#2,"<ENTER>";          [2318]
460 a$="":WHILE a$="":a$=INKEY$:WEND:IF a$     [4246]
<>CHR$(13) THEN 410
470 LOCATE#2,12,1:PRINT#2,"         ":FOR t=   [5096]
1 TO 3:IF t=i THEN 480 ELSE LOCATE#2,2,2+t
:PRINT#2,"                 "
480 NEXT                                       [350]
490 b=0                                        [438]
500 ON i GOTO 540,770,920                      [1332]
510 '                                          [117]
520 '                                          [117]
530 '********************************          [1285]
540 '*          Block lesen          *         [978]
550 '********************************          [1285]
560 '                                          [117]
570 '                                          [117]
580 POKE &4D0C,&A1:POKE &4D18,&A1:CALL &4D     [3100]
1E:CALL &4D00
590 GOSUB 630:GOTO 410                         [1160]
600 '                                          [117]
610 '                                          [117]
620 '********************************          [1285]
630 '*      Header auswerten         *         [1565]
640 '********************************          [1285]
650 '                                          [117]
660 '                                          [117]
670 CLS#1:PRINT#1,"Name : ";:FOR i=&4E00 T     [11104]
O &4E0F:PRINT#1,CHR$(PEEK(i));:NEXT:LOCATE
 #1,25,1:PRINT#1,"Block: ";PEEK(&4E10):he=
&4E00:IF PEEK(he+23)<>0 THEN LOCATE #1,40,
1:PRINT#1,"First Block
680 IF PEEK(he+17)<>0 THEN LOCATE#1,40,1:P     [3944]
RINT#1,"Last Block "
690 LOCATE#1,1,2:PRINT#1,"File  Loc.: &";H     [16246]
EX$(PEEK(he+22)*256+PEEK(he+21)-(PEEK(he+1
6)-1)*&800,4);:LOCATE #1,20,2:PRINT#1,"Fil
e  len.: &";HEX$(PEEK(he+25)*256+PEEK(he+2
4),4);:LOCATE #1,40,2:PRINT#1,"Boot : &";H
EX$(PEEK(he+27)*256+PEEK(he+26),4);
700 LOCATE#1,1,3:PRINT#1,"Block Loc.: &";H     [11065]
EX$(PEEK(he+22)*256+PEEK(he+21),4);:LOCATE
 #1,20,3:PRINT#1,"Block len.: &";HEX$(PEEK
(he+20)*256+PEEK(he+19),4);:LOCATE #1,40,3
:PRINT#1,"Blocks :";INT( PEEK(he+25)*256+P
EEK(he+24)-1)/2048)+1
710 LOCATE#1,1,4:PRINT#1,"Type : &X";BIN$(     [12569]
PEEK(he+18),8);" &";HEX$(PEEK(he+18),2);"
";:IF PEEK(he+18)=0 THEN PRINT#1,"Basic" E
LSE IF PEEK(he+18)=1 THEN PRINT#1,"Protect
ed Basic"
720 IF PEEK(he+18)=2 THEN PRINT#1,"Binary"     [5196]
 ELSE IF PEEK(he+18)=8 THEN PRINT#1,"Ascii
"
730 RETURN                                     [555]
740 '                                          [117]
750 '                                          [117]
760 '********************************          [1285]
770 '*         Block sichern         *         [2234]
780 '********************************          [1285]
790 '                                          [117]
800 '                                          [117]
810 LOCATE #1,1,5:PRINT#1,"          Speed (   [3835]
0), (1) oder (2)";SPC(10);
820 a$="":WHILE a$="":a$=INKEY$:WEND:IF a$     [4179]
<"0" OR a$>"2" THEN 820
830 IF a$="2" THEN CALL &4D2C ELSE SPEED W     [3054]
RITE VAL(a$)
840 LOCATE#1,1,5:PRINT#1,STRING$(54,32);       [2221]
850 LOCATE #1,16,5:PRINT#1,"Press REC and      [5069]
PLAY then any key:":WHILE INKEY$="":WEND
860 LOCATE #1,16,5:PRINT#1,"                   [2336]
                   "
870 POKE &4D0C,&9E:POKE &4D18,&9E:CALL &4D     [2758]
00
880 GOTO 410                                   [450]
890 '                                          [117]
900 '                                          [117]
910 '********************************          [1285]
920 '*           Aendern             *         [824]
930 '********************************          [1285]
940 '                                          [117]
950 '                                          [117]
960 CLS#2:PRINT#2,"    Aendern":PRINT#2:FOR    [6321]
 i=1 TO 3:LOCATE#2,2,i+2:PRINT#2,mu$(i):NE
XT
970 a$="":WHILE a$="":a$=INKEY$:WEND           [2835]

980 IF a$<"1" OR a$>"3" THEN 970               [2671]
990 i=VAL(a$):LOCATE#2,2,2+i:PRINT#2,CHR$(     [3015]
24)mu$(i)CHR$(24)
1000 LOCATE #2,12,1:PRINT#2,"<ENTER>";         [2318]
1010 a$="":WHILE a$=""!a$=INKEY$:WEND:IF a     [4246]
$<>CHR$(13) THEN 410
1020 LOCATE#2,12,1:PRINT#?,"         ":FOR t   [5599]
=1 TO 3:IF t=i THEN 1030 ELSE LOCATE#2,2,2
+t:PRINT#2,"                 "
1030 NEXT                                      [350]
1040 hd=-1                                     [754]
1050 ON i GOTO 1090,1200,410                   [1711]
1060 '                                         [117]
1070 '                                         [117]
1080 '********************************         [1285]
1090 '*        Header Editor          *        [988]
1100 '********************************         [1285]
1110 '                                         [117]
1120 '                                         [117]
1130 CLS:hd=-1:GOSUB 1350:LOCATE#1,1,5:PRI     [3761]
NT#1,"<E>dit <M>enue";
1140 a$="":WHILE a$="":a$=INKEY$:WEND:a$=U     [7975]
PPER$(a$):IF a$="M" THEN LOCATE#1,1,5:PRIN
T#1,SPC(56);:GOTO 920
1150 IF a$<>"E" THEN 1140                      [1441]
1160 GOSUB 1470:GOSUB 630:GOTO 1090            [2165]
1170 '                                         [117]
1180 '                                         [117]
1190 '********************************         [1285]
1200 '*       Programm Editor         *        [995]
1210 '********************************         [1285]
1220 '                                         [117]
1230 '                                         [117]
1240 CLS:hd=0:GOSUB 1350                       [1177]
1250 LOCATE#1,1,5:PRINT#1,"<E>ditor <M>enu     [6727]
e  <"CHR$(240)">  <"CHR$(241)"> ";"Seite "
;b;
1260 a$="":WHILE a$="":a$=INKEY$:WEND:a$=U     [7975]
PPER$(a$):IF a$="M" TH N LOCATE#1,1,5:PFIN
T#1,SPC(56);:GOTO 920
1270 IF a$=CHR$(240) THEN IF b>0 THEN b=b-     [3579]
1:CLS:GOSUB 1350
1280 IF a$=CHR$(241) THEN IF b<7 THEN b=b+     [4147]
1:CLS:GOSUB 1350
1290 IF a$>="0" AND a$<="7" THEN b=VAL(a$)     [5329]
:CLS:GOSUB 1350:GOTO 1250
```

```
1300 IF a$<>"E" THEN 1250                     [1435]
1310 GOSUB 1470:GOTO 1200                     [2321]
1320 '                                        [117]
1330 '                                        [117]
1340 '********************************        [1285]
1350 '*          Dump                  *      [549]
1360 '********************************        [1285]
1370 '                                        [117]
1380 '                                        [117]
1390 'hd;Flag fuer Header oder Pgm:b=Seite    [2520]
 im Block 0-7
1400 IF hd THEN s=0:p=&4E00 ELSE s=PEEK(he    [4555]
+22)*256+PEEK(he+21)+b*256:p=&5000+b*256
1410 FOR i=0 TO 255 STEP 16:PRINT"&";HEX$(    [8123]
i+s,4);"  ";:FOR t=0 TO 15:PRINT HEX$(PEEK
(p+t+i),2);" ";:IF t=7 THEN PRINT"  ";
1420 NEXT:PRINT"   ";:FOR t=i TO i+15:PRIN    [3776]
T CHR$(1)CHR$(PEEK(t+p));:NEXT:PRINT:NEXT

1430 RETURN                                   [555]
1440 '                                        [117]
1450 '                                        [117]
1460 '********************************        [1285]
1470 '*        Editor                  *      [427]
1480 '********************************        [1285]
1490 '                                        [117]
1500 '                                        [117]
1510 x=0:y=0:xx=8:LOCATE 8,1:PRINT#1," <TA    [4781]
B> Exit";:CALL &BB81
1520 a$="":WHILE a$="":a$=INKEY$:WEND         [2835]
1530 IF a$=CHR$(9) THEN RETURN                [1524]
1540 IF a$=CHR$(240) THEN IF y>0 THEN y=y-    [1599]
1
1550 IF a$=CHR$(241) THEN IF y<15 THEN y=y    [1223]
+1
1560 IF a$=CHR$(242) THEN IF x>0 THEN x=x-    [3051]
1:GOSUB 1650
1570 IF a$=CHR$(243) THEN IF x<31 THEN x=x    [2733]
+1:GOSUB 1650
1580 LOCATE xx,y+1                            [1218]
1590 IF x>15 THEN GOSUB 1840:GOTO 1520 ELS    [3707]
E GOSUB 1750:IF NOT fl THEN GOTO 1520
1600 fl=0:rr=r:PRINT a$;:WHILE fl=0:a$=INK    [9838]
EY$:GOSUB 1750:WEND:PRINT a$;:r=rr*16+r:PO
KE p+(16*y+x),r:LOCATE x+61,y+1:PRINT CHR$
(1)CHR$(r);
1610 IF x<> 15 THEN a$=CHR$(243):GOTO 1570    [5876]
 ELSE x=0:GOSUB 1650:a$=CHR$(241):GOTO 154
0
1620 '                                        [117]
1630 '                                        [117]
1640 '********************************        [1285]
1650 '*     Berechnung von XX          *      [1241]
1660 '********************************        [1285]
1670 '                                        [117]
1680 '                                        [117]
1690 IF x<16 THEN xx=3*x+8:IF x>7 THEN xx=    [3056]
xx+2
1700 IF x>15 THEN xx=x+45                     [512]
1710 RETURN                                   [555]
1720 '                                        [117]
1730 '                                        [117]
1740 '********************************        [1285]
1750 '*      Editor Input              *      [1286]
1760 '********************************        [1285]
1770 '                                        [117]
1780 '                                        [117]
1790 a$=UPPER$(a$):fl=0:IF (a$>="0" AND a$    [4469]
<="9") OR (a$>="A" AND a$<="F") THEN r=VAL
("&"+a$):fl=-1
1800 RETURN                                   [555]
1810 '                                        [117]
1820 '                                        [117]
1830 '********************************        [1285]
1840 '*       Ascii Input              *      [1963]
1850 '********************************        [1285]
1860 '                                        [117]
1870 '                                        [117]
1880 IF a$>CHR$(239) OR a$="" OR a$<CHR$(3    [1497]
2) THEN RETURN
1890 a=ASC(a$):POKE p+((x-16)+16*y),a:PRIN    [2979]
T a$;
1900 l=3*(x-16)+8:IF (x-16)>7 THEN l=l+2      [1379]
1910 LOCATE l,y+1:PRINT HEX$(a,2);            [2500]
1920 IF x<31 THEN x=x+1 ELSE IF y<15 THEN     [3140]
x=16:y=y+1
1930 GOSUB 1650                               [825]
1940 LOCATE xx,y+1                            [1218]
1950 RETURN                                   [555]
```

Echtzeit-Uhr selbstgebaut



Natürlich kann man eine Uhr auch per Software realisieren, doch wer will das Uhrenprogramm schon ununterbrochen laufen lassen, um z.B. für einen Terminplaner immer die korrekte Zeit zur Verfügung zu haben? Viele Anwender, die ihren Computer vorwiegend für Terminplanung und organisatorische Aufgaben benutzen, schätzen darum sehr den Zugriff auf Echtzeit.

Besonders interessant wird diese Option für CPC 6128-Anwender, weil CP/M Plus über einige Funktionen verfügt, in denen das Datum und die Uhrzeit erforderlich sind (DATE, DATE CONTINOUS, INITDIR). Weitere Gründe für die Notwendigkeit einer Echtzeit-Uhr ergeben sich, wenn der Computer mit Prozeßsteuerungen betraut ist, die zeitabhängig arbeiten. Ein Beispiel hierfür wäre eine Heizungssteuerung mit Nachtabsenkung oder z.B. die berühmte automatische Fischfütterung im Urlaub. Diese und andere Gründe mögen den Leser bewegen, sich für die hier dargestellte Selbstbauanleitung zu interessieren. Bei der folgenden Beschreibung wurde nicht nur auf eine unkomplizierte Struktur der erforderlichen Elektronik Wert gelegt, sondern es werden konkrete Hinweise zur oft lästigen aber notwendigen Mechanik geliefert. Die Bauanleitung schließt daher das Thema Gehäusekonstruktion mit ein. Auch hierbei wurde darauf geachtet, den mechanischen Aufwand so gering wie möglich zu gestalten. Dennoch soll nicht verschwiegen werden, daß der interessierte Nachbauer über Grundkenntnisse der notwendigen Arbeitstechniken verfügen muß (zumindest löten und feilen). Doch nun zur Sache, zunächst in Form von Highlights zu der Bastelei.

- Funktionen der Uhr:

Uhrzeit in Stunden, Minuten und

Sekunden,
Datum in Tag, Monat und Jahr, sowie Wochentag.
Uhrzeit kann dabei im landesüblichen 24-Std.-Modus ausgegeben werden.
- Automatische Schaltjahrkorrektur.
- Versorgung der Uhr über eingebauten Akku bei ausgeschaltetem Computer (Gangreserve 3 Monate).
- Automatisches Nachladen des Akkus bei eingeschaltetem Computer (Ladezeit für völlig entleerten Akku: 14 Std.).
- Passende Adapter für alle Schneider-CPC-Computer.

Tabelle 1

Position der	Drahtbrücke	I/O-Adressen des Uhren-Moduls
A	FBE2	für Adressregister
	FBE3	für Datenregister
B	FBE6	für Adressregister
	FBE7	für Datenregister
C	FBEA	für Adressregister
	FBEB	für Datenregister
D	FBEE	für Adressregister
	FBEF	für Datenregister
E	FBF2	für Adressregister
	FBF3	für Datenregister
F	FBF6	für Adressregister
	FBF7	für Datenregister
G	FBFA	für Adressregister
	FBFB	für Datenregister
H	FBFE	für Adressregister
	FBFF	für Datenregister

Tabelle 1: Auswahl der I/O-Adressen

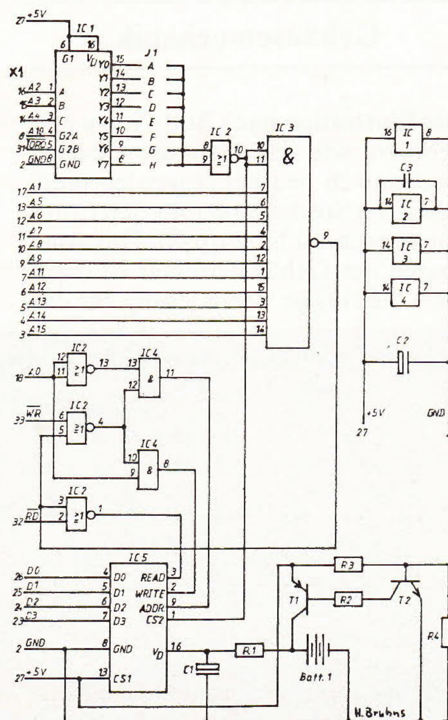


Bild 1: Der Schaltplan der Uhr

- Installation des Moduls am Erweiterungsstecker ohne Eingriff in den Computer.
- Einfache Programmierung der Uhr, sogar aus dem BASIC-Direct-Command-Mode.
- Die Uhr läuft auch fehlerfrei weiter, wenn sie vollständig vom Computer getrennt wird.
- Geringe Größe (100*50*25 mm).
- Niedriger Preis (im Selbstbau unter DM 100,-).

Schaltungsbeschreibung

Im Schaltplan nach Bild 1 ist die gesamte Elektronik des Uhrenzusatzes dargestellt. Kernstück des Ganzen ist ein integrierter Schaltkreis des japanischen Herstellers Suwa Seikosha, nämlich RTC-58321 Real Time Clock Module. Dieser IC (im Schaltplan IC1) beinhaltet alles, was die vorstehenden Highlights zur Uhrenfunktion ausmacht. Der IC ist für eine direkte Kommunikation mit einem Mikroprozessor vorgesehen, und kann sich bei Ausfall der Versorgungsspannung "einigeln", d.h., die Zeit läuft intern weiter, während die Ein- und Ausgänge zum angeschlossenen Mikroprozessor abgetrennt werden. Einziges Erfordernis ist eine Notstromversorgung. Zu diesem Zweck ist Batt. 1 vorhanden. Das Netzwerk aus T1, T2, C1, R1, R2, R3 und R4 schaltet dabei zwischen Notstrom und Computerstrom um, und veranlaßt das Nachladen von Batt.1, wenn der Computer eingeschaltet ist. Die Verbindung zum Computer stellen die IC's IC2, IC3, IC4 und IC5 dar. Mit ihrer Hilfe erfolgt das Decodieren eines zulässigen Ein-/Ausgabe-Adressbereichs. Zu diesem Schaltungsteil gehört auch das Feld für Lötbrükken. Durch wahlweises Setzen des Jumpers (J1) kann der Benutzer zwischen acht verschiedenen Basisadressen für den Uhrenzusatz wählen. Diese Wahlmöglichkeit ist nützlich, wenn man zusätzliche Peripherie an seinem Computer betreibt und deshalb eventuell eine Doppelbelegung von E/A-Adressen umgangen werden muß. Das Layout der Platine kann so ausgelegt werden, daß anstelle der Drahtbrücke als Jumper ein Achtfach-DIL-Schalter eingesetzt werden kann. Die Zuordnung von Jumper-Position und selektiertem Adressbereich ist Tabelle 1 zu entnehmen.
Was "Adressregister" und "Datenregister in Tabelle 1 bedeuten, wird im Software-Teil dieser Bauanleitung detailliert beschrieben.
Besonders erwähnenswert ist auch

noch der Steckverbinder, über den die Uhr mit dem Computer verbunden wird. Die gewählte Bauform des Steckers gestattet den Anschluß der Uhr an alle CPC's, indem auf diesen Stecker wahlweise ein Adapter für die 464/664, bzw. 6128 Expansion-Connectors aufgesetzt werden kann.

Das Handwerkliche

Wer dem Vorschlag des Autors bezüglich des verwendeten Gehäuses folgen will, kann der Maßskizze nach Bild 2 entnehmen, wie groß die verwendete Platine werden darf, welche Aussparungen zu berücksichtigen sind und an welcher Stelle Bohrungen für die Befestigungsschrauben vorzusehen sind. Der Autor hat sich vorher davon überzeugt, daß die Elektronik auf der verbleibenden Fläche Platz findet, auch wenn man die Uhr auf einer Experimentierplatine aufbaut.
Eine freie Verdrahtung auf einer Experimentierplatine erfüllt durchaus die notwendigen Voraussetzungen zum einwandfreien Funktionieren der Uhr, wie der Prototyp des Autors beweist. Die Experimentierversion läuft seit mehr als einem halben Jahr ohne Beanstandung, obwohl sie zu Demonstrationszwecken schon häufig herumgereicht (verliehen) und an unterschiedlichen CPC's betrieben wurde.
Einzige Besonderheit bei der Montage der Bauteile ist der Akku. Die Bauhöhe des Akkus erfordert, daß er gewissermaßen halb in die Platinenebene abgesenkt werden muß. Aus diesem Grunde weist die Platine den Ausschnitt auf, der am rechten Rand der Maßzeichnung nach Bild 2 zu sehen ist. Auch die Position des Steckers muß entsprechend der im Bild 2 dargestellten Bemaßung gewählt werden, wenn die Platine in das vorgeschlagene Gehäuse passen soll. Als Alternative für alle Freunde großer Verarbeitungsqualität ist eine gedruckte Schaltung gemäß Bild 2 denkbar. Auf Anfrage ist beim Verlag die Quelle für diese Platine zu erhalten.
Besondere Aufmerksamkeit wurde bei dieser Bastelei dem Gehäuse geschenkt. Einem selbstgebauten Zusatz muß ja nicht unbedingt anzusehen sein, daß hier "die Axt im Haus" den Zimmermann erspart hat.
Hochwertige Elektronik gehört nun einmal nicht in eine Seifendose oder in Opas letzte Zigarren-Holzkiste. Es sind genügend Industriegehäuse im Handel, die ein ansprechendes Design besitzen, einfach zu beschaffen und

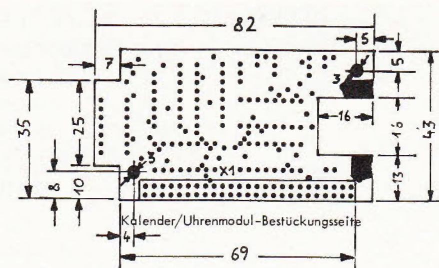


Bild 2: Layoutvorschlag der Platine

preiswert sind. Die Daten des verwendeten Gehäuses sind, wie alle anderen wichtigen Daten zu den Bauteilen, der Tabelle 2 entnehmen.

Gehäusemechanik

Der Illustration nach Bild 4 ist zu entnehmen, wie das Gehäuse-Oberteil mechanisch bearbeitet werden muß, damit der Steckadapter für den Computeranschluß hineinpaßt. Das Aussparen des Gehäusedeckels ist dann auch die einzige Veränderung, die das Gehäuse erfahren muß. Zu diesem Zweck eignet sich eine Laubsäge mit Sägeblatt für Metallbearbeitung. Die

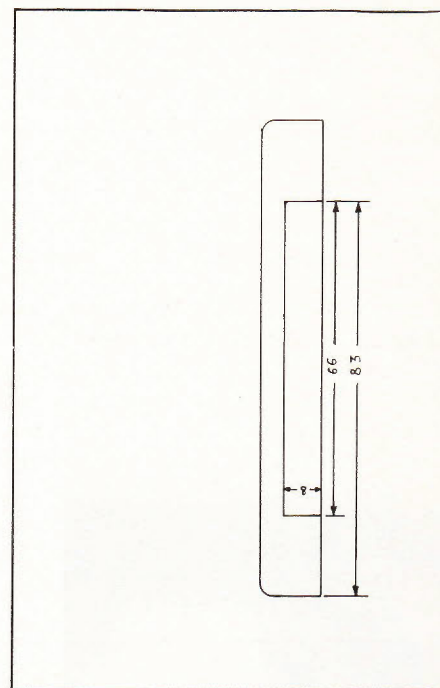


Bild 4: So wird der Schlitz im Gehäuse angebracht

Stückliste für das Uhren-Modul

Bauteil-Nr.	Bezeichnung	Beschreibung
IC 1	8-Kanal Demultiplexer	SN74LS138 (Texas Instruments)
IC 2	vierfach NOR	SN74LS02 (Texas Instruments)
IC 3	NAND	SN74LS133 (Texas Instruments)
IC 4	vierfach AND	SN74LS08 (Texas Instruments)
IC 5	Uhr/Kalender	RTC-58321 (Suwa Seikosha)
R 1	Widerstand	100 Ohm, 0.25 W
R 2	Widerstand	10 kOhm, 0.125 W
R 3	Widerstand	51 kOhm, 0.125 W
R 4	Widerstand	10 kOhm, 0.125 W
T 1	Transistor	2N2907
T 2	Transistor	2N2222 (Texas Instruments)
C 1	Tantal-Kondensator	4.7 uF, 10V
C 2	Tantal-Kondensator	22 uF, 25V
C 3	Abblock-Kondensator	0.1 uF bipolar
Batt. 1	NiCd-Akkumulator	3.6 V 3/100 DKO (Varta)
X 1	Stiftstecker	flach, offen, 50-polig (z.B. Speedy G08 ITT-Cannon)
ohne	Federleiste	für Stiftstecker X 1, 50-polig (z.B. Speedy G80 ITT-Cannon)
ohne	Flachkabel	50-polig, Raster 1.27mm Länge ca. 50 mm
ohne	Gehäuse	Flachgehäuse Normalausführung 100x50x25 mm (z.B. OKW Best. Nr. 9010087 o. BOPLA Best.Nr. 05410000)
ohne	Platine	gedruckte Schaltung oder Experimentierplatine (Maße wie in Bild 2)
ohne	2 Stck. Schrauben	Linsenblechschraube 2.9x6.5 mm mit Kreuzschlitz DIN 7981 (zur Befestigung der Platine)
	nur CPC 6128 !	
ohne	Flachkabel-Stecker	Delta-Ribbon 50-polig, verriegelbar (z.B. Scotchflex 3M)
	nur CPC 464/CPC 664 !	
ohne	Flachkabel-Stecker	Steckkarten-Verbinder, Raster 2.54 mm, 50-polig, ohne Flansch (z.B. Speedy G03 ITT-Cannon)

Tabelle 2: Stückliste für das Uhren-Modul

Kontur des Ausschnitts wird damit grob vorgearbeitet. Danach kann man mit einer kleinen Feile die "Wogen" glätten, sowie die Ecken ausarbeiten. Etwas Sorgfalt sollte man bei dieser Arbeit walten lassen, weil ihre Ausführung sichtbar bleibt. Korrekte Ausführung dieser Arbeit, sowie korrektes Positionieren des Steckers auf der Uhrenplatine hat zur Folge, daß das geschlossene Gehäuse relativ staubdicht bleibt und daß der aufgesteckte Adapter eine mechanische Zugentlastung erhält.
Hält man sich an die Bemaßung der Platine, dann paßt sie ohne Probleme in die untere Gehäuseschale und kann hier mit zwei M3-Schrauben fixiert werden. Die obere Gehäuseschale wird von unten zugeschraubt, nachdem sie von oben aufgesetzt wurde (siehe Bild 5). Eine Belüftung des Gehäuses ist auch bei Dauerbetrieb nicht erforderlich, weil die Verlustleistung der Zusatz-Hardware minimal ist.

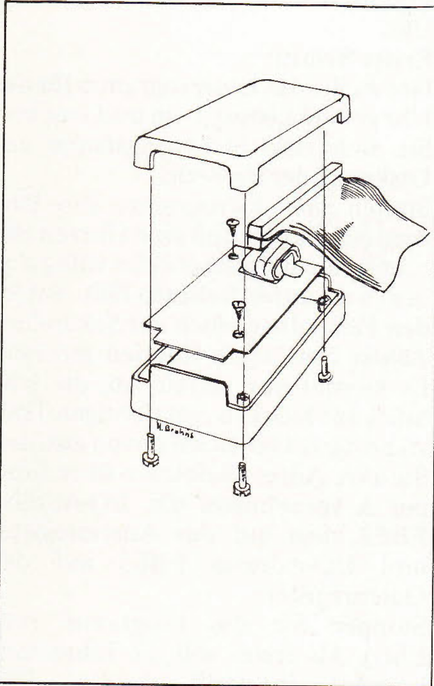


Bild 5: Der Zusammenbau des Gehäuses

Die Adapter

In Bild 6 sind die beiden Adapter dargestellt, mit denen die Uhr an den Computer angeschlossen wird. Die Bauteileliste (Tabelle 2) gibt Aufschluß darüber, welche Bauteile zu den unterschiedlichen Versionen für CPC 464/664 und CPC 6128 erforderlich sind. Die Adapterkabel sind in Quetschtechnik gefertigt. Beim Quetschen ist darauf zu achten, daß die Lage der Stecker auf dem Kabel, sowie die Orientierung der Stecker korrekt sind. Bild 6 gibt Hinweise zur richtigen Montage der Quetschstecker.

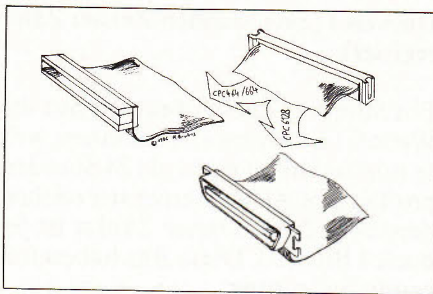


Bild 6: Steckerauswahl für 464/664/6128

Der weiche Part

Der Hardware-Teil ist relativ unproblematisch, wenn man sich vergegenwärtigt, daß keinerlei Angleicharbeiten anfallen. Der Hersteller des Uhrenbausteins hat werkseitig dafür gesorgt, daß der Uhrenoszillator auf der richtigen Frequenz arbeitet, um die Langzeitgenauigkeit zu erreichen, die wir von Seiko-Uhren gewohnt sind. Der Rest ist Digitaltechnik konventionellster Art. Ein wenig Arbeit bleibt jedoch bei der softwaremäßigen Installation der Uhr. Den CP/M Plus-Anwendern sei an dieser Stelle gesagt, daß der vorliegende Artikel noch keine Betriebssystem-Modifikation zum Ersatz der "alten" Uhrenroutinen des CP/M Plus enthält. Der Autor ist jedoch dabei, diese Option vorzubereiten, um sie zum Gegenstand eines speziellen Beitrags zu machen. Im folgenden Teil des Artikels ist eine Lösung dargestellt, um die Uhr unter Basic zu betreiben. Es sind zwei Möglichkeiten notwendig, nämlich das Stellen der Uhr und das Auslesen. Das Grundprinzip zum Lesen und Schreiben ist gleich. Als erstes muß das Adressregister der Uhr mit der Adresse des internen Zählerregisters adressiert werden, danach kann dieses interne Zählerregister gelesen oder beschrieben werden.

Auslesen der Uhr

Um die Uhr schnell und sicher auszulesen, ist eine kurze Maschinenroutine erforderlich, die als DATA-Zeilen im Listing 1 steht.
Sie ist für eine Basisadresse von &FBE2 geschrieben. Sollte es notwendig sein, eine andere Adresse zu wählen, so ist unter Berücksichtigung von Tabelle 1 lediglich ein Eintrag aus zwei Bytes in der DATA-Zeile 590 des Basic-Listings anzupassen. Es ist die Stelle, die in der Z80-Assemblerzeile 37 die Basisadresse der Uhr bestimmt. In DATA-Zeile 590 sind es die Positionen 6 und 7. Bitte beachten Sie, daß die Basisadresse mit dem LSB beginnt (E2,FB anstatt FB,E2)! Aus dem Z80-Listing ist ebenfalls entnehmbar, auf welche absolute Adresse im Basic-Programm die Ergebnisse der Ausleseroutine geschrieben werden. Die Zeilen 65 bis 77 im Z80-Listing stellen den Zusammenhang zwischen Bedeutung des Uhrenregisters und der RAM-Adresse dar. So erhält man z.B. bei einem PEEK von &A53E die Zehnerstelle des Monatszählers oder bei einem PEEK von &A548 den Sekunden-Einer, nachdem man mit CALL &A500 das Leseprogramm gestartet hat. Man kann die Uhrenregister natürlich auch direkt über die Basic-Befehle INP und OUT beeinflussen, um an die gewünschte Information zu gelangen. Jedoch hat die Uhr einige Tücken, die an geeigneter Stelle in einem weiteren Beitrag erläutert werden sollen.
Zunächst soll der Hinweis genügen, daß die abgedruckte Basic-Routine ein korrektes Lesen der Uhr garantiert.

Stellen der Uhr

Die Umkehrung des Auslesens ist das Beschreiben der Uhrenregister. Wenn man bedenkt, daß es nur einmal erforderlich ist, die Uhr zu stellen, sofern kein technischer Defekt das Wechseln des Akkus erforderlich macht, und daß man nur zweimal pro Jahr die Uhrzeit um jeweils eine Stunde verändern muß, wenn man eine Sommerzeit-Anpassung wünscht, ist eine Luxus-Schreibroutine für die Uhr recht überflüssig. Von dieser Überlegung ausgehend soll hier eine Methode vorgestellt werden, mit der man die Uhr im Direct-Command-Mode einstellen kann. Dazu ist es erforderlich, einige Informationen zu den Uhrenregistern zu liefern.

Die internen Register der Uhr: 1. Das Adressregister

In Tabelle 1 dieses Beitrags wurden die möglichen Basisadressen des Uhrenmoduls vorgestellt. Hier taucht erstmals der Begriff "Adressregister" auf. Dieses Register ist nur beschreib-

HARDWARE

bar (write only). In diesem Register vermerkt man, mit welchem Zählerregister der Uhr man Daten austauschen will. Wenn man das Adressregister beschrieben hat, gilt die Adresse, die man dort eingetragen hat, für alle weiteren Lese- oder Schreiboperationen auf das Datenregister der Uhr, d.h., ein Zählerregister wird über das Adressregister ständig selektiert, bis man den Inhalt des Adressregisters ändert. Doch was sind Zählerregister?

2. Die Zählerregister

Vergegenwärtigen Sie sich, daß eine Uhr im Prinzip nur aus einer Aneinanderreihung von Zählern besteht, die sich ihre Überträge "intelligent" weiterreichen. Der Sekunden-Einer-Zähler z.B. zählt immer von 0 bis 9 und gibt dann einen Impuls auf den Sekunden-Zehner-Zähler, der wiederum einen Übertrag auf den Minuten-Einer-Zähler liefert, wenn er bis 5 gezählt hat. Komplizierter werden die Überträge bei den Tageszählern, weil ihr maximaler Zählerstand vom Monatszähler abhängt (es gibt nun mal keinen 31. Februar). Der Februar ist für alle Uhren mit Datumsanzeige ein richtiger Breaker, weil hier auch noch berücksichtigt werden muß, ob Schaltjahr ist oder nicht. Der langen Schreiberei Sinnlosigkeit ist, daß man auf alle diese Zähler im Innern der Uhr zugreifen kann, indem man sie über das mittlerweile bekannte Adressregister adressiert und dann über das Datenregister Daten einschreibt oder ausliest. In der Tabelle 3 ist aufgelistet, welchen Wert man in das Adressregister schreiben muß, um mit welchem Zählerregister "sprechen" zu können.

Die Besonderheiten

Hinweis 1 (zum Stunden-Zehner Zählregister)

Ein Stunden-Zehner braucht nur die Werte 0,1,2 annehmen zu können, weil es nunmal nicht mehr als 24 Stunden pro Tag gibt. Als Zählerregister reichen demnach 2 Bit. Dieser Zähler ist jedoch 4 Bit breit. Diese Bits haben folgende Bedeutung:
Bit 0 und Bit 1:
enthalten den Zählerstand.
Bit 3=0:
Stunden werden im 12 Stunden-Rhythmus ausgegeben.
Bit 3=1:
Stunden werden im 24 Stunden-Rhythmus ausgegeben (landesüblich).
Bit 2:
immer 0, wenn Uhr im 24 Stunden-Rhythmus arbeitet.
Bit 2=0:
vormittags, wenn im 12 Stunden-Rhythmus
Bit 2=1:
nachmittags, wenn im 12 Stunden-Rhythmus.

Hinweis 2 (zum Wochentag-Zählregister)

Der Wochentag-Zähler läuft modulo 7 (immer von 0 bis 6). Es bleibt Ihnen persönlich überlassen, welchem Zählerstand Sie welchen Wochentag zuordnen. Im Beispielprogramm nach Listing 1 wurde der Zählerstand 0 dem Montag zugeordnet. Sie müssen beim Stellen dieses Registers nur dafür sorgen, daß Sie den Wert einschreiben, der Ihrer Wochentag-Zuordnung für den konkreten Wochentag entspricht (logisch).

Hinweis 3 (zum Tage-Zehner Zählregister)

In diesem Register wird festgelegt, in welchem Schaltjahres-Modus die Uhr laufen soll.
Bit 0 und bit 1:
enthalten den Zählerstand.
Bit 3:
muß 0 sein
Bit 2=0:
Schaltjahr-Korrektur nach Gregorianischem Kalender
Bit 2=1:
Schaltjahr-Korrektur nach Showa Kalender

Mit diesem Rüstzeug ist es Ihnen nun möglich, Ihrer Uhr den richtigen Zeitgeist einzuhauchen. Damit Sie nach all den verwirrenden Beschreibungen jedoch nicht völlig hilflos sind, ist im folgenden Abschnitt ein Beispiel aufgezeigt.

Praktisches Beispiel zum Stellen der Uhr.
Erster Schritt:
Geben Sie das Leseprogramm für die Uhr gemäß Listing 1 ein und machen Sie sich eine Sicherungskopie auf Diskette oder Kassette.
Starten Sie das Programm. Ihre Uhr wird ein Datum und eine Uhrzeit anzeigen, die möglicherweise völlig aus dem bekannten Rahmen fällt. Auf jeden Fall sollte jedoch der Sekunden-Zähler brav seine Runden machen. Es kommt nun darauf an, die Uhr Stück für Stück in den richtigen Trab zu bringen. Gehen wir davon aus, daß Sie Ihre Adress-Selektion über Jumper A vornehmen, d.h. I/O-Adresse FBE2 zeigt auf das Adressregister und I/O-Adresse FBE3 auf das Datenregister.
Stoppen Sie das Programm (mit ESC). Als erstes soll die Jahreszahl korrekt eingestellt werden. Der Jahres-Zehner muß (zum Erscheinungszeitpunkt dieses Artikels) auf 8 zeigen. Um das zu erreichen, muß zunächst eben dieser Zähler zum Beschreiben vorbereitet werden. Man erreicht diesen Zähler, indem man in das Adressregister der Uhr den Wert C (hexadezimal) einschreibt und das geht so:
OUT &FBE2,&C (Return)
Damit ist der Jahres-Zehner scharf.
Nun muß er mit einer 8 beschrieben werden, nämlich wie folgt:
OUT &FBE3,8 (Return)
Machen wir gleich den Jahres-Einer hinterdrein:
OUT &FBE2,&B
OUT &FBE3,6
Wenn Sie nun Ihr Leseprogramm er-

Tabelle 3

Inhalt des Adressregisters	selektiertes Zählerregister	Zählbereich	Besonderheiten
0	Sekunden-Einer	0 bis 9	keine
1	Sekunden-Zehner	0 bis 5	keine
2	Minuten-Einer	0 bis 9	keine
3	Minuten-Zehner	0 bis 5	keine
4	Stunden-Einer	0 bis 9	keine
5	Stunden-Zehner	siehe Hinweis 1	siehe Hinweis 1
6	Wochentag	0 bis 6	siehe Hinweis 2
7	Tage-Einer	0 bis 9	keine
8	Tage-Zehner	0 bis 3	siehe Hinweis 3
9	Monate-Einer	0 bis 9	keine
A	Monate-Zehner	0 bis 1	keine
B	Jahre-Einer	0 bis 9	keine
C	Jahre-Zehner	0 bis 9	keine
D bis F	selektiert verschiedene Kontrollfunktionen, die in dieser Applikation von untergeordneter Bedeutung sind		

Tabelle 3: Anwahl der Zählerregister

neut starten, sollten Sie zu Ihrer Freude feststellen, daß die Jahreszahl richtig gestellt ist. In entsprechender Manier verfahren Sie bitte mit allen anderen Registern, wobei Sie die Hinweise zu Schaltjahr-Korrektur und 12/24 Stunden-Darstellung berücksichtigen wollen.

Wenn Sie Freude an Ihrem Uhren-Modul finden, können Sie sich selbstverständlich eine elegante Uhren-Stell-Routine schreiben. Der Verlag ist sicherlich an der Veröffentlichung interessiert. Desgleichen sind nun fantasiebegabte Programmierer gefordert, die Uhr in nützliche Programme einzubinden.

Termin-Planer, Statusanzeigen, grafische Aufbereitung als "Analoguhr" mit Zifferblatt und Zeigern (und vielleicht auch Pendel) sind einige Anregungen, die der Autor beizusteuern hat. Wir sind alle gespannt auf Ihre Ideen! (Dipl.-Ing. H. Bruhns/ME)

```
10 d$(0)="MONTAG, "                                   [922]
20 d$(1)="DIENSTAG, "                                 [1012]
30 d$(2)="MITTWOCH, "                                 [1022]
40 d$(3)="DONNERSTAG, "                               [1361]
50 d$(4)="FREITAG, "                                  [342]
60 d$(5)="SAMSTAG, "                                  [1646]
70 d$(6)="SONNTAG, "                                  [916]
80 MODE 1                                             [506]
90 R1=&X1100110                                       [863]
100 R2=&X0                                            [91]
110 R3=&X1100110                                      [732]
120 R4=&X1100110                                      [333]
130 R5=&X1100110                                      [484]
140 R6=&X1100110                                      [720]
150 R7=&X111100                                       [414]
160 R8=&X0                                            [313]
170 SYMBOL 240,R1,R2,R3,R4,R5,R6,R7,R8                [2382]
180 MEMORY &A4FF                                      [463]
190 GOSUB 490                                         [883]
200 CALL &A500                                        [617]
210 PRINT "          BEISPIELPROGRAMM F"+CHR          [2308]
$(240)+"R DAS"
220 PRINT "        ECHTZEIT-KALENDER/UHRENMO          [2526]
DUL"
230 PRINT                                             [361]
240 PRINT "                      RCC 8509"            [1129]
250 LOCATE 3,11                                       [709]
260 CALL &A500                                        [617]
270 t=PEEK(&A542)                                     [618]
280 PRINT d$(t);                                      [648]
290 PRINT USING "#";PEEK(&A540);                      [1679]
300 PRINT USING "#";PEEK(&A541);                      [1340]
310 PRINT ".";                                        [665]
320 PRINT USING "#";PEEK(&A53E);                      [1700]
330 PRINT USING "#";PEEK(&A53F);                      [1696]
340 PRINT ".";                                        [665]
350 PRINT USING "#";PEEK(&A53C);                      [1676]
360 PRINT USING "#";PEEK(&A53D);                      [1704]
370 PRINT "     ";                                    [612]
380 PRINT USING "#";PEEK(&A543);                      [1683]
390 PRINT USING "#";PEEK(&A544);                      [1695]
400 PRINT ":";                                        [521]
410 PRINT USING "#";PEEK(&A545);                      [1675]
420 PRINT USING "#";PEEK(&A546);                      [1655]
430 PRINT ":";                                        [521]
440 PRINT USING "#";PEEK(&A547);                      [1699]
450 PRINT USING "#";PEEK(&A548);                      [1520]
460 LOCATE 3,11                                       [709]
470 GOTO 260                                          [417]
480 END                                               [110]
490 'LOADER                                           [949]
500 RESTORE                                           [621]
510 FOR I=&A500 TO &A53B                              [777]
520 READ H$                                           [495]
530 H=VAL("&"+H$)                                     [1030]
540 POKE I,H                                          [150]
550 NEXT I                                            [375]
560 RETURN                                            [555]
570 'GETDATIM                                         [818]
580 DATA 11,3C,A5,21,2F,A5,06,0D,CD,12,A5,            [2639]
A6,12,23,13,10,F7,C9 'MAIN
590 DATA 78,D5,C5,5F,01,E2,FB,7B,3D,ED,79,            [6055]
03,ED,78,57,7B,3D,0B,ED,79,03,ED,78,BA,20,
EA,C1,D1,C9 'SAVEX
600 DATA 0F,0F,1,0F,3,0F,7,3,0F,7,0F,7,0F             [2135]
'MASK
```

Dis-K-Patcher V 3.0
Programmbeschreibung



Read Sector:
Lesen eines beliebigen Sektors von Diskette. Dieser Sektor kann nach dem Einlesen direkt mit dem Full-Screen-Editor bearbeitet werden. Dabei kann zwischen ASCII-Eingabe oder Hexadezimaler-Eingabe gewählt werden.

Write Sector:
Ein gerade eingelesener und überarbeiteter Sektor wird wieder zurück auf Diskette geschrieben.

Searching for Data:
Dieser Befehl kann angewendet werden, um auf der Diskette nach irgend einem Text zu suchen. Die maximale Textlänge beträgt 20 Zeichen. Wird der Text gefunden, so wird sofort in den Full-Screen-Editor (siehe 'Read Sector') verzweigt.

Wipe Disk:
"Säubert" die Diskette. Dabei wird die gesamte Diskette, gelöscht. Nach Ausführung hat die Diskette genau den gleichen Zustand wie nach dem Formatieren (alle Bytes sind mit &e5 belegt).

Unerase Files:
Dies ermöglicht, gelöschte Files zu "retten". Dabei wird nicht nur geprüft, ob der Eintrag noch in der Directory steht, sondern auch, ob die Sektoren bzw. Blöcke dieses zu rettenden Programms nicht schon von einem anderen Programm belegt werden. Wäre dies nämlich der Fall, so wäre eine Rettung unmöglich.

Information:
Gibt alle wichtigen Daten über ein File auf der Diskette z.B. Name, Typ, Startadresse etc. auf dem Bildschirm aus. Diese Informationen können direkt nach Belieben verändert werden. Besonders interessant dabei ist die Möglichkeit, ein File zu verstecken (Hidden) oder gegen Überschreiben zu schützen (Read Only). Dabei muß einfach bei "Hidden" bzw. "Read Only" die Zahl 1 eingetragen werden, dann ist dieser File versteckt bzw. geschützt. Bei eintragen von der Zahl 0 wird dieser Zustand wieder aufgehoben.

Print Information:
Gibt die wichtigen Daten aller Files auf dem Drucker aus.

Sort Dir:
Dadurch kann das Directory, also das Inhaltsverzeichnis der Diskette, sortiert werden. Besonders interessant ist dies für jene User, die oft unter CP/M arbeiten, da es dort nämlich einen solch komfortablen Befehl wie CAT, der gleich alle Einträge sortiert auf den Bildschirm bringt, nicht gibt. Weiter bringt dies auch einen Vorteil für jene, die unter Basic öfter mit dem |DIR-Befehl arbeiten, da dieser Befehl durch die Möglichkeit mit Wildcards zu arbeiten, im Prinzip wesentlich leistungsfähiger als der CAT-Befehl ist.

Pack Disk:

Dieser Befehl bewirkt folgendes:

1. Das Directory wird sortiert (nach den Namen).
2. Einträge von gelöschten Files in der Directory werden endgültig gelöscht.
3. Die Blockverteilung jeden Programms wird optimiert. Dies bringt eine erhebliche Verkürzung der Ladezeiten und Ordnung auf der Diskette. Dieser Vorgang darf nicht abgebrochen werden. Hat man also einmal "Pack Disk" angewählt, so muß man warten, bis der Computer fertig ist. Ein gewaltsamer Abbruch, z.B. durch "ESC" oder einen Reset, hat einen vollkommenen Verlust des Disketteninhalts zur Folge.
4. Der verbleibende Teil der Diskette wird gesäubert (siehe "Wipe Disk").

Block Partition:
Zeigt an, wie die Diskette physisch belegt ist, also welche Blöcke der Diskette mit Daten belegt sind.

Backup:
Zieht eine Sicherheitskopie der eingelegten Diskette auf Band.

Restore:
Eine mit "Backup" erstellte Sicherheitskopie wird wieder auf Diskette zurückgeschrieben. Dabei ist zu beachten, daß der gesamte Inhalt der eingelegten Diskette überschrieben wird.

Alles weitere Wichtige, was man während des Programmlaufs wissen sollte, wird immer im rechten Fenster "EDITOR" eingeblendet.
Ich wünsche Ihnen nun viel Spaß und Erfolg mit dem Dis-K-Patcher.

(Stefan R. Mayer)

```
100  CLEAR:GOSUB 8000                            [830]
102  GOSUB 4900                                  [950]
1000 GOSUB 3200                                  [953]
1007 GOSUB 3000                                  [961]
1008 GOSUB 4900                                  [950]
1009 brokenflag%=false%:nobreak%=true%           [3629]
1010 tast$=UPPER$(INKEY$):IF tast$="" THEN [2898]
 1010
1015 IF tast$=ubreak$ THEN MODE 2:PRINT me [3578]
ssage$;", active for";INT((TIME-active
 )/300);"seconds":PRINT:CLEAR:END
1020 cur%=INSTR("RWSEUIDPTLAO",tast$):IF c [3762]
ur%=0 THEN 1010
1025 l%=LEN(cur$(cur%)):help=INT((50-l%)/2 [2544]
)+1
1030 CLS:PRINT TAB(help);cur$(cur%):PRINT  [4393]
TAB(help-1);STRING$(l%+2,154):PRINT
1050 ON cur% GOSUB 2000,2100,2300,2900,240 [4417]
0,2500,3500,3600,3800,4100,4600,4700
1060 IF foundflag%=true% AND cur%=3 THEN G [2891]
OSUB 2011
1099 GOTO 1000                                   [339]
2000 REM "--- Lesen ---"                         [1186]
2005 GOSUB 3250                                  [1174]
2010 GOSUB 6000:CLS#4                            [1241]
2011 '                                           [117]
2012 GOSUB 2800                                  [907]
2013 POKE modus,readbl%:GOSUB 5900               [1352]
2015 IF foundflag%=false% THEN page%=0           [1898]
2017 GOSUB 7000                                  [867]
2020 GOSUB 5000                                  [883]
2022 readflag%=true%                             [1001]
2025 GOSUB 2850                                  [905]
2030 GOSUB 5600                                  [954]
2099 RETURN                                      [555]
2100 REM "--- Schreiben ---"                     [1912]
2101 GOSUB 3250                                  [1174]
2103 IF readflag%=false% THEN LOCATE 18,4: [9905]
PRINT CHR$(7);CHR$(24);
 " No Sector Read ";CHR$(24):FOR o=1 TO 30
00:NEXT o:GOTO 2199
2104 GOSUB 6000:CLS#4:GOSUB 4900                 [2232]
2105 LOCATE 15,11:PRINT sure$                    [1646]
2106 tast$=UPPER$(INKEY$):IF tast$="" THEN [2922]
 2106
2107 ON INSTR("NY",tast$)+1 GOTO 2106,2199 [2583]
,2108
2108 LOCATE 15,11:PRINT CHR$(20)                 [1561]
2109 GOSUB 2800                                  [907]
2110 FOR spalte%=0 TO 31                         [555]
2115   FOR zeile%=0 TO 31 STEP 2                 [974]
2120     POKE bufadr+zeile%\2+spalte%*16,V [5209]
AL("&"+MID$(hc$(spalte%),zeile%
 +1,2))
2130   NEXT zeile%                               [571]
2131 NEXT spalte%                                [448]
2140 POKE modus,writebl%:GOSUB 5900              [1715]
2145 GOSUB 7000                                  [867]
2150 GOSUB 2850                                  [905]
2199 RETURN                                      [555]
2200 REM "-- Format der Disk ermitteln --- [2278]
"
2201 WINDOW SWAP 4,0:PEN 1:LOCATE 1,12:|DI [2425]
SC
2202 errflag%=false%:IF INSTR("AB",drive$) [4853]
=0 THEN errflag%=true%:GOTO 2295
2203 POKE ident(0),&FF:POKE ident(1),&FF:P [1692]
OKE &BE78,&FF
2205 ON ERROR GOTO 2250:|DRIVE,@drive$           [2740]
2207 IF PEEK(ident(0))=&FF AND drive$="A"  [3423]
THEN errflag%=true%:GOTO 2295
2208 IF PEEK(ident(1))=&FF AND drive$="B"  [3303]
THEN errflag%=true%:GOTO 2295
2210 drive%=ASC(drive$)-65:form%=PEEK(iden [6100]
t(drive%)):format%=form%\65:IF format%
 =0 THEN format%=3
2220 GOSUB 5900                                  [1053]
2230 GOTO 2295                                   [344]
2250 errflag%=true%:RESUME 2295                  [1766]
2295 POKE &BE78,&0:ON ERROR GOTO 0:PEN 0:W [2939]
INDOW SWAP 4,0
2299 RETURN                                      [555]
2300 REM "--- Nach Zeichenkette Suchen --- [1639]
"
2301 GOSUB 3250:LOCATE 5,10:PRINT CHR$(24) [7402]
;" Data to Search ";CHR$(24);" : ";
 namebak$:LOCATE 1,4
2303 GOSUB 4300:GOSUB 6000                       [1993]
2305 PRINT:PRINT                                 [743]
2308 x%=23:y%=10:eg$=name$:ml%=20:stat%=an [4909]
umeric%:GOSUB 6100
2309 IF brokenflag%=true% THEN 2399              [765]
2310 GOSUB 3400                                  [1009]
2311 GOSUB 2800                                  [907]
2315 GOSUB 3200                                  [953]
2320 help$=eg$:GOSUB 4500:eg$=help$:name$= [4038]
eg$:IF namebak$<>eg$ THEN foundflag%=
 false%
2321 namebak$=eg$                                [910]
2322 help2$=SPACE$(150):help2=@help2$:hb%= [5424]
PEEK(help2+2):lb%=PEEK(help2+1)
2323 IF foundflag%=true% THEN sektor%=sekt [3872]
or%+1:GOSUB 4300
2324 foundflag%=false%:help$(5)=""               [1920]
2325 POKE modus,readbl%:bufadr=sadr+50           [1525]
2327 WHILE foundflag%=false% AND track%<40 [2743]
2335   WHILE sektor%<=maxsek%(format%)           [3444]
2337     IF INKEY$=ubreak$ THEN GOSUB 4800 [1253]
:GOTO 2399
2340     GOSUB 7000                              [867]
2350     FOR buffer=bufadr TO bufadr+511 S [1814]
TEP 128
2355       POKE help2+1,buffer-256*INT(buf [1915]
fer/256)
2357       POKE help2+2,INT(buffer/256)          [1226]
2358       helpa=1+(buffer-bufadr)\128:hel [4398]
p$(helpa)=help2$
2360       IF INSTR(help2$,eg$) THEN found [4508]
flag%=true%:page%=(helpa-1)\2:GOTO
 2378
2365     NEXT buffer                             [1340]
```

ANWENDUNG

```
2370       IF INSTR(MID$(help$(5),119,20)+LE [7556]
FT$(help$(1),20),eg$) THEN page%=1:
      sektor%=sektor%-1:foundflag%=true%:GO
SUB 4300:GOTO 2378
2372       help$(5)=help$(4)               [299]
2373       sektor%=sektor%+1               [2455]
2374 WEND                                  [390]
2375 sektor%=1:track%=track%+1             [1784]
2378 WEND                                  [390]
2380 GOSUB 2850                            [905]
2385 IF foundflag%=false% THEN 2397        [1080]
2386 CLS:LOCATE 19,5:PRINT CHR$(7);CHR$(24 [4104]
);"  Found '";eg$;"'   ";CHR$(24);
2397 POKE help2+1,lb%:POKE help2+2,hb%:hel [1982]
p2$=SPACE$(150)
2399 RETURN                                [555]
2400 REM "--- Recover ---"                 [846]
2401 GOSUB 6200:CLS#4                      [1755]
2403 GOSUB 2800                            [907]
2405 GOSUB 2600                            [851]
2406 nofile%=true%:ccflag%=false%          [2460]
2407 FOR i=1 TO hdir                       [1096]
2408    IF LEFT$(hdir$(i),1)>CHR$(15) THEN [2051]
nofile%=false%
2409 NEXT i                                [375]
2410 IF nofile%=true% THEN LOCATE 13,5:PRI [7313]
NT CHR$(7);CHR$(24);
 " No Erased File(s) Recovered ";CHR$(24);
:FOR i=1 TO 2000:NEXT i:GOTO 2499
2415 LOCATE 15,5:PRINT CHR$(24);" Found "; [2822]
CHR$(24);" :"
2425 FOR i=1 TO hdir                       [1096]
2427    IF MID$(hdir$(i),13,1)<>CHR$(0) OR [4424]
LEFT$(hdir$(i),1)<=CHR$(15) THEN 2496
2430    file$=MID$(hdir$(i),2,11):LOCATE 25 [5553]
,5:hname$=file$:GOSUB 4200:PRINT
      hname$;
2435    block$=""                          [689]
2437    FOR o=1 TO hdir                    [1346]
2439       IF INSTR(hdir$(o),file$) THEN blo [3220]
ck$=block$+MID$(hdir$(o),17,16)
2441    NEXT o                             [369]
2443    recoverflag%=true%                 [2108]
2444    LOCATE 15,7:PRINT CHR$(20);"Checkin [2473]
g ..."
2445    FOR o=1 TO LEN(block$)             [1928]
2447       help$=MID$(block$,o,1):IF help$=C [2079]
HR$(0) THEN 2460
2449          FOR p=1 TO hdir              [1109]
2451             IF INSTR(MID$(hdir$(p),17,16) [4796]
,help$) AND LEFT$(hdir$(p),1)<=CHR$
         (15) THEN recoverflag%=false%
2453          NEXT p                       [364]
2460    NEXT o                             [369]
2464    m$="Successful":IF recoverflag%=fal [4704]
se% THEN m$="Not "+m$
2470    LOCATE 15,7:PRINT CHR$(7);CHR$(20); [2758]
m$:GOSUB 4900
2473    IF recoverflag%=false% THEN 2495   [3253]
2474    LOCATE 15,9:PRINT "UnErase ? (Y/N)" [2089]
2475    tast$=UPPER$(INKEY$):IF tast$="" TH [2921]
EN 2475
2476    ON INSTR("NY",tast$)+1 GOTO 2475,24 [2671]
95,2477
2477    FOR o=1 TO hdir                    [1346]
2478       IF INSTR(hdir$(o),file$) THEN MID [2936]
$(hdir$(o),1,1)=CHR$(0)
2479    NEXT o                             [369]
2480    ccflag%=true%                      [539]
2495    LOCATE 15,9:PRINT CHR$(20)         [1015]
2496 NEXT i                                [375]
2497 IF ccflag%=true% THEN GOSUB 2700      [1818]
2499 RETURN                                [555]
2500 REM "--- Informationen ueber Files -- [2641]
-"
2501 ccflag%=false%:retflag%=true%:jumpfla [4314]
g%=true%:printflag%=false%:nobreak%=
  false%
2502 'Einsprung Druck                      [2106]
2503 GOSUB 6200:CLS#4                      [1755]
2504 IF printflag%=false% THEN GOSUB 3350  [2194]
2506 PRINT TAB(5);CHR$(24);" Name      ";C [6828]
HR$(24);" :";TAB(30);CHR$(24);
 " Extension ";CHR$(24);" :":PRINT
2507 PRINT TAB(5);CHR$(24);" Type      ";C [1497]
HR$(24);" :":PRINT
2508 PRINT TAB(5);CHR$(24);" Start     ";C [4107]
HR$(24);" :";TAB(30);CHR$(24);
 " Length    ";CHR$(24);" :":PRINT
```

```
2509 PRINT TAB(5);CHR$(24);" Call      ";C [7452]
HR$(24);" :";TAB(30);CHR$(24);
 " User      ";CHR$(24);" :":PRINT
2512 PRINT TAB(5);CHR$(24);" Hidden    ";C [5982]
HR$(24);" :";TAB(30);CHR$(24);
 " Read Only ";CHR$(24);" :":PRINT:PRINT
2514 PRINT TAB(5);CHR$(24);" Files     ";C [4462]
HR$(24);" :";TAB(30);CHR$(24);
 " Current   ";CHR$(24);" :";
2515 GOSUB 2600:file=1:hfile=1:total=0:ron [4062]
file%=0:hidfile%=0
2516 LOCATE 19,15:PRINT USING u0$;diranz   [3100]
2517 IF printflag%=true% AND hdir<>0 THEN  [3886]
GOSUB 4115
2518 WHILE hfile<=hdir                     [1192]
2519    IF MID$(hdir$(hfile),13,1)<>CHR$(0) [5161]
 OR LEFT$(hdir$(hfile),1)>CHR$(15)
   THEN hfile=hfile+1:GOTO 2592
2520    cflag%=false%:LOCATE 44,15:PRINT US [3985]
ING u0$;file
2521    POKE modus,readbl%:bufadr=hbuf:bl=2 [7154]
*ASC(MID$(hdir$(hfile),17,1)):
   track%=FNtrack%(bl):sektor%=FNsektor%(b
l):GOSUB 7000
2523    type%=PEEK(bufadr+18):start=PEEK(bu [9036]
fadr+21)+256*PEEK(bufadr+22):length=
   PEEK(bufadr+24)+256*PEEK(bufadr+25):cal
=PEEK(bufadr+26)+256*PEEK(bufadr      +
27):total=total+length
2524    name$=MID$(hdir$(hfile),2,8):extent [4910]
$=MID$(hdir$(hfile),10,3):help$=name$+
   extent$
2525    user%=ASC(LEFT$(hdir$(hfile),1))   [1938]
2526    type$=type$(3):IF type%<=2 THEN typ [3391]
e$=type$(type%)
2529    hidden%=0:help=ASC(MID$(extent$,2,1 [7201]
)):IF help AND 128 THEN hidden%=1:MID$
   (extent$,2,1)=CHR$(help-128):hidfile%=h
idfile%+1
2530    readon%=0:help=ASC(MID$(extent$,1,1 [7357]
)):IF help AND 128 THEN readon%=1:MID$
   (extent$,1,1)=CHR$(help-128):ronfile%=r
onfile%+1
2535    LOCATE 19,4:PRINT name$:LOCATE 44,4 [3121]
:PRINT extent$
2537    LOCATE 19,6:PRINT type$            [777]
2540    LOCATE 19,8:PRINT USING u0$;start:L [5751]
OCATE 44,8:PRINT USING u0$;length
2542    LOCATE 19,10:PRINT USING u0$;cal:LO [5185]
CATE 44,10:PRINT USING u0$;user%
2544    LOCATE 19,12:PRINT USING u0$;hidden [2249]
%:LOCATE 44,12:PRINT USING u0$;readon%
2545    IF printflag%=true% THEN GOSUB 4400 [3162]
:GOTO 2589
2550    stat%=anumeric%:eg$=name$:x%=19:y%= [7475]
4:xh%=0:ml%=8:GOSUB 6100:name$=eg$:IF
   pageflag%=true% THEN 2570 ELSE IF broke
nflag%=true% THEN 2569
2555    stat%=anumeric%:eg$=extent$:x%=44:y [7517]
%=4:xh%=0:ml%=3:GOSUB 6100:extent$=eg$
   :IF pageflag%=true% THEN 2570 ELSE IF b
rokenflag%=true% THEN 2569
2557    stat%=numeric%:eg$=STR$(start):x%=1 [9087]
9:y%=8:xh%=0:ml%=5:GOSUB 6100:start=
   VAL(eg$):IF start<0 OR start>65535 THEN
 PRINT CHR$(7);:GOTO 2557
2558    IF pageflag%=true% THEN 2570 ELSE I [3401]
F brokenflag%=true% THEN 2569
2559    stat%=numeric%:eg$=STR$(length):x%= [7587]
44:y%=8:xh%=0:ml%=5:GOSUB 6100:length=
   VAL(eg$):IF length<0 OR length>65535 TH
EN PRINT CHR$(7);:GOTO 2559
2560    IF pageflag%=true% THEN 2570 ELSE I [3401]
F brokenflag%=true% THEN 2569
2561    stat%=numeric%:eg$=STR$(cal):x%=19: [8609]
y%=10:xh%=0:ml%=5:GOSUB 6100:cal=VAL
   (eg$):IF cal<0 OR cal>65535 THEN PRINT
CHR$(7);:GOTO 2561
2562    IF pageflag%=true% THEN 2570 ELSE I [3401]
F brokenflag%=true% THEN 2569
2563    stat%=numeric%:eg$=STR$(user%):x%=4 [9514]
4:y%=10:xh%=3:ml%=2:GOSUB 6100:user%=
   VAL(eg$):IF user%<0 OR user%>15 THEN PR
INT CHR$(7);:GOTO 2563
2564    IF pageflag%=true% THEN 2570 ELSE I [3401]
F brokenflag%=true% THEN 2569
2565    stat%=numeric%:eg$=STR$(hidden%):x% [9730]
=19:y%=12:xh%=4:ml%=1:GOSUB 6100:
```

```
    hidden%=VAL(eg$):IF hidden%<>0 AND hidd
en%<>1 THEN PRINT CHR$(7);:GOTO        2
565
2566   IF pageflag%=true% THEN 2570 ELSE I [3401]
F brokenflag%=true% THEN 2569
2567   stat%=numeric%:eg$=STR$(readon%):x% [9728]
=44:y%=12:xh%=4:ml%=1:GOSUB 6100:
   readon%=VAL(eg$):IF readon%<>0 AND read
on%<>1 THEN PRINT CHR$(7);:GOTO        2
567
2569   IF cflag%=false% AND brokenflag%=tr [3000]
ue% THEN 2593
2570   IF cflag%=false% THEN 2589 [1357]
2571   ccflag%=true%:LOCATE 10,17:PRINT CH [2241]
R$(7);
2572   PRINT CHR$(24);" Data Has Been Chan [5547]
ged, Write ? (Y/N) ";CHR$(24);
2573   htast$=UPPER$(INKEY$):IF htast$="" [3419]
THEN 2573
2574   ON INSTR("YN",htast$)+1 GOTO 2573,2 [2547]
575,2588
2575   POKE bufadr+21,start-256*INT(start/ [11276]
256):POKE bufadr+22,INT(start/256):
   POKE bufadr+24,length-256*INT(length/25
6):POKE bufadr+25,INT(length/256):       P
OKE bufadr+26,cal-256*INT(cal/256):POKE bu
fadr+27,INT(cal/256)
2576   check=0 [177]
2577   FOR i=bufadr TO bufadr+66:check=che [2709]
ck+PEEK(i):NEXT i
2578   POKE bufadr+67,check-256*INT(check/ [3424]
256):POKE bufadr+68,INT(check/256)
2579   POKE modus,writebl%:GOSUB 7000 [1590]
2580   MID$(extent$,1,1)=CHR$(ASC(MID$(ext [3060]
ent$,1,1))+readon%*128)
2582   MID$(extent$,2,1)=CHR$(ASC(MID$(ext [3020]
ent$,2,1))+hidden%*128)
2584   hdir$(hfile)=CHR$(user%)+name$+exte [4829]
nt$+MID$(hdir$(hfile),13,20)
2585   FOR i=1 TO hdir [1096]
2586     IF INSTR(hdir$(i),help$) THEN MID [4472]
$(hdir$(i),2,11)=MID$(hdir$(hfile),2,
   11)
2587   NEXT i [375]
2588   LOCATE 10,17:PRINT CHR$(20):IF brok [2526]
enflag%=true% THEN 2593
2589   IF pageflag%=false% OR pageflag%=tr [5250]
ue% AND tast$=CHR$(249) THEN hfile=
   hfile+1:file=file+1 GOTO 2592
2590   IF pageflag%=true% AND tast$=CHR$(2 [7147]
48) THEN hfile=hfile-1:file=file-1:IF
   hfile<1 THEN hfile=1:file=1
2591   IF pageflag%=true% AND tast$=CHR$(2 [7501]
48) AND hfile<>1 AND MID$(hdir$(hfile)
   ,13,1)<>CHR$(0) THEN hfile=hfile-1:GOTO
2591
2592 WEND [390]
2593 IF ccflag%=false% THEN 2599 [1051]
2594 GOSUB 2700 [911]
2599 RETURN [555]
2600 REM "--- Directory Einladen ---" [2097]
2602 POKE modus,readbl%:GOSUB 3400 [1327]
2605 track%=dir%(format%):FOR sektor%=1 TO [5963]
   4:bufadr=hbuf+(sektor%-1)*512:GOSUB
   7000:NEXT sektor%
2635 ptr=1 [256]
2636 FOR i=hbuf TO hbuf+2016 STEP 32 [2844]
2637   IF PEEK(i+3)=229 THEN 2660 [1937]
2638   FOR o=0 TO 31:hdir$(ptr)=hdir$(ptr) [3624]
+CHR$(PEEK(o+i)):NEXT o
2655   ptr=ptr+1 [303]
2660 NEXT i [375]
2665 hdir=ptr-1 [650]
2675 diranz=0 [466]
2680 FOR i=1 TO hdir [1096]
2685   IF MID$(hdir$(i),13,1)=CHR$(0) AND [6332]
LEFT$(hdir$(i),1)<=CHR$ (15) THEN
   diranz=diranz+1
2690 NEXT i [375]
2695   bufadr=sadr+50 [1661]
2699 RETURN [555]
2700 REM "--- Directory Abspeichern ---" [2754]
2701 GOSUB 2800 [907]
2702 IF hdir=0 THEN 2799 [465]
2704 FOR i=hdir+1 TO 64:hdir$(i)=STRING$(3 [3196]
2,229):NEXT i
2705 ptr=1 [256]
2707 FOR p=hbuf TO hbuf+2016 STEP 32:FOR o [4620]
=0 TO 31:POKE p+o,ASC(MID$(hdir$(ptr),
 o+1,1)):NEXT o:ptr=ptr+1:NEXT p
2710 POKE modus,writebl%:track%=dir%(forma [7125]
t%):FOR sektor%=1 TO 4:bufadr=hbuf+
(sektor%-1)*512:GOSUB 7000:NEXT sektor%
2715 bufadr=sadr+50 [1661]
2720 GOSUB 2850 [905]
2799 RETURN [555]
2800 REM "--- Busy Andrucken ---" [1298]
2810 LOCATE#1,71,1:PRINT#1,CHR$(24);" Busy [2456]
 ";CHR$(24);
2849 RETURN [555]
2850 REM "--- Busyanzeige Loeschen ---" [1871]
2860 LOCATE#1,71,1:PRINT#1,CHR$(20); [1371]
2899 RETURN [555]
2900 REM "--- Diskette 'wipen' ---" [1192]
2901 IF wipeflag%=true% THEN 2905 [2570]
2902 GOSUB 6200:CLS#4:GOSUB 4900:LOCATE 15 [3875]
,5:PRINT sure$
2903 tast$=UPPER$(INKEY$):IF tast$="" THEN [2961]
 2903
2904 ON INSTR("NY",tast$)+1 GOTO 2903,2999 [3103]
,2905
2905 LOCATE 19,4:PRINT CHR$(20);CHR$(24);" [4440]
 Wiping Disk ";CHR$(24);
2906 POKE modus,writebl%:bufadr=hbuf:FOR i [5420]
=hbuf TO hbuf+1023:POKE i,229:NEXT i
2907 IF wipeflag%=false% THEN ziel%=0 [1778]
2909 FOR i=FNtrack%(ziel%*2)*maxsek%(forma [7616]
t%)-1+FNsektor%(ziel%*2) TO maxsek%
   (format%)*39+maxsek%(format%)-1
2910    sektor%=i MOD maxsek%(format%)+1:tr [4974]
ack%=i\maxsek%(format%)
2915    GOSUB 7000 [867]
2920 NEXT i [375]
2925 bufadr=sadr+50:wipeflag%=false% [2524]
2999 RETURN [555]
3000 REM "--- Menue 1 ---" [761]
3005 CLS [91]
3010 PRINT TAB(22);" Options" [1294]
3015 PRINT TAB(22);STRING$(9,154) [814]
3020 PRINT [361]
3023 PRINT TAB(5);CHR$(24);" R ";CHR$(24); [4270]
"   ";cur$(1);TAB(26);CHR$(24);" S ";
 CHR$(24);"   ";cur$(3)
3024 PRINT [361]
3025 PRINT TAB(5);CHR$(24);" W ";CHR$(24); [4431]
"   ";cur$(2);TAB(26);CHR$(24);" E ";
 CHR$(24);"   ";cur$(4)
3026 PRINT [361]
3027 PRINT TAB(5);CHR$(24);" D ";CHR$(24); [5696]
"   ";cur$(7);TAB(26);CHR$(24);" P ";
 CHR$(24);"   ";cur$(8)
3028 PRINT [361]
3029 PRINT TAB(5);CHR$(24);" I ";CHR$(24); [3612]
"   ";cur$(6);TAB(26);CHR$(24);" T ";
 CHR$(24);"   ";cur$(9)
3035 PRINT:PRINT TAB(5); CHR$(24);" U ";CHR [7205]
 $(24);"   ";cur$(5);TAB(26);CHR$(24);
  " L ";CHR$(24);"   ";cur$(10)
 3040 PRINT :PRINT TAB(5);CHR$(24);" A ";CH [5248]
 R$(24);"   ";cur$(11);TAB(26);CHR$(24)
  ;" O ";CHR$(24);"   ";cur$(12)
 3050 PRINT:PRINT:PRINT TAB(20);"Please cho [4071]
 ose  ";CHR$(143);
 3099 RETURN [555]
3100 REM "--- Ed 1 ---" [683]
3102 CLS#4 [386]
3103 PRINT#4,TAB(3);"'DEL' Exit" [1640]
3105 PRINT#4,TAB(4);CHR$(240);"  Cursor Up [2489]
"
3107 PRINT#4,TAB(4);CHR$(241);"  Cursor Do [2814]
wn"
3110 PRINT#4,TAB(4);CHR$(242);"  Cursor Le [2575]
ft
3113 PRINT#4,TAB(4);CHR$(243);"  Cursor Ri [2928]
ght"
3115 PRINT#4,TAB(4);CHR$(160);CHR$(241);" [3168]
PageDown"
3117 PRINT#4,TAB(4);CHR$(160);CHR$(240);" [2597]
PageUp"
3125 PRINT#4,TAB(4);CHR$(160);"A AlphaNume [3432]
ric Input"
3130 PRINT#4,TAB(4);CHR$(160);"H HexaDecim [3435]
al  Input"
3135 PRINT#4,TAB(4);CHR$(160);"P Print Inf [3384]
ormation"
3149 RETURN [555]
```

```
3200 REM "--- Ed 3 ---"                              [1008]
3205 CLS#4                                           [386]
3210 PRINT#4,TAB(3);"'DEL'  Exit"                    [1709]
3249 RETURN                                          [555]
3250 REM "--- Ed 4 ---"                              [715]
3255 CLS#4                                           [386]
3260 PRINT#4,TAB(2);"'ENTER' Next InputMas          [2606]
k";
3265 PRINT#4,TAB(2);"'CLR'   Del  Char Lef          [3648]
t"
3299 RETURN                                          [555]
3300 REM "--- Statusanzeige Anum/Hexa ---"          [3754]
3305 LOCATE #4,3,12:PRINT#4,"Current : ";            [1989]
3310 IF status%=anumeric% THEN PRINT#4,"Al          [4142]
phaNumeric"
3315 IF status%=hexa%      THEN PRINT#4,"He         [4830]
xaDecimal "
3349 RETURN                                          [555]
3350 REM "--- Ed 5 ---"                              [817]
3355 CLS#4                                           [386]
3357 PRINT#4,TAB(2);"'DEL'   EXIT"                   [1287]
3358 PRINT#4                                         [315]
3360 PRINT#4,TAB(2);"'ENTER' Next InputMas          [2606]
k";
3365 PRINT#4,TAB(2);"'CLR'   Del  Char Lef          [3889]
t";
3370 PRINT#4,TAB(3);CHR$(160);CHR$(241);"           [2016]
    PageDown"
3375 PRINT#4,TAB(3);CHR$(160);CHR$(240);"           [1717]
    PageUp"
3399 RETURN                                          [555]
3400 REM "--- Clean --"                              [1686]
3405 readflag%=false%                                [1026]
3410 ERASE hdir$,ch$,hc$,help$:DIM hdir$(6          [4346]
4),ch$(31),hc$(31),help$(5)
3449 RETURN                                          [555]
3499 RETURN                                          [555]
3500 REM "--- Sortieren Des Inhaltsverzeic          [5603]
hnisses --"
3501 GOSUB 6200                                      [899]
3502 GOSUB 3200                                      [953]
3503 GOSUB 4900                                      [950]
3504 CLS#4                                           [386]
3505 PRINT                                           [361]
3510 PRINT TAB(5);CHR$(24);" Sort ";CHR$(2          [7031]
4);" :    ";CHR$(24);" N ";CHR$(24);
 "    Name      ";CHR$(24);" E ";CHR$(24);"
    Extension"
3515 tast$=UPPER$(INKEY$):IF tast$="" THEN          [3089]
 3515
3520 wahl=INSTR("NE"+ubreak$,tast$):IF wah          [2562]
l=0 THEN 3515
3522 IF wahl=3 THEN GOSUB 4800:GOTO 3599            [2315]
3525 help$="Names":beg=2:l=8:IF wahl=2 THE          [3376]
N help$="Extensions":beg=11:l=3
3526 PRINT CHR$(11);CHR$(18);TAB(8);CHR$(2          [3681]
4);" Sorting ";CHR$(24);" : ";help$
3527 GOSUB 2800                                      [907]
3528 GOSUB 2600                                      [851]
3540 GOSUB 3700:IF brokenflag%=true% GOTO           [3330]
3590
3545 GOSUB 2700                                      [911]
3590 GOSUB 2850                                      [905]
3599 RETURN                                          [555]
3600 REM "--- Diskette Reorganisieren ---"          [2180]
3601 IF backupflag%=false% THEN GOSUB 6200          [2258]
3602 CLS#4                                           [386]
3605 GOSUB 2800                                      [907]
3610 packflag%=true%:GOSUB 3800                      [2498]
3611 FOR i=1 TO hdir                                 [1096]
3612   IF LEFT$(hdir$(i),1)>CHR$(15) THEN           [3860]
hdir$(i)=STRING$(32,229)
3613 NEXT i                                          [375]
3615 beg=2:l=12:GOSUB 3700                           [1317]
3620 ziel%=2                                         [454]
3625 FOR i9=1 TO hdir                                [1580]
3626   IF hdir$(i9)=STRING$(32,229) THEN 3          [2266]
691
3628   LOCATE 17,14:PRINT CHR$(24);" Packi          [6087]
ng ";CHR$(24);" : ";:hname$=MID$(hdir$
   (i9),2,11):GOSUB 4200:PRINT hname$
3630     FOR i5=17 TO 32                             [1212]
3635       block=ASC(MID$(hdir$(i9),i5,1)):I        [3497]
F block=0 THEN 3690
3640       block$=CHR$(block):ziel$=CHR$(zie        [2048]
l%)
3650       htrack%(1)=FNtrack%(block*2):hsek        [12412]
tor%(1)=FNsektor%(block*2):htrack%(2)=
      htrack%(1):hsektor%(2)=hsektor%(1)+1:
IF hsektor%(2)>maxsek%(format%)
 THEN hsektor%(2)=1:htrack%(2)=htrack%(2)+
1
3655       htrack%(3)=FNtrack%(ziel%*2):hsek        [9181]
tor%(3)=FNsektor%(ziel%*2):htrack%(4)=
      htrack%(3):hsektor%(4)=hsektor%(3)+1:
IF hsektor%(4)>maxsek%(format%)
 THEN hsektor%(4)=1:htrack%(4)=htrack%(4)+
1
3657       help=block-2:LOCATE 17+help MOD 1        [2618]
8,help\18+3:PRINT CHR$(143);
3658       help=ziel%-2:LOCATE 17+help MOD 1        [4344]
8,help\18+3:PRINT CHR$(24);CHR$(232);
      CHR$(24);
3660       IF htrack%(1)=htrack%(3) AND hsek        [3360]
tor%(1)=hsektor%(3) THEN 3689
3665       POKE modus,readbl%:FOR i=1 TO 4:b        [7679]
ufadr=hbuf+(i-1)*512:track%=htrack%(i)
      :sektor%=hsektor%(i):GOSUB 7000:NEXT
i
3672       POKE modus,writebl%                       [1385]
3675       FOR i=3 TO 4:bufadr=hbuf+(i-3)*51        [5870]
2:track%=htrack%(i):sektor%=hsektor%
      (i):GOSUB 7000:NEXT i
3680       FOR i=1 TO 2:bufadr=hbuf+(i+1)*51        [5514]
2:track%=htrack%(i):sektor%=hsektor%
      (i):GOSUB 7000:NEXT i
3685       FOR i=1 TO hdir:help$=MID$(hdir$(        [10189]
i),17,16):help=INSTR(help$,ziel$):IF
      help THEN MID$(help$,help,1)=block$:M
ID$(hdir$(i),17,16)=help$
3687       NEXT i                                    [375]
3688       MID$(hdir$(i9),i5,1)=ziel$                [1963]
3689       ziel%=ziel%+1                             [547]
3690     NEXT i5                                     [441]
3691 NEXT i9                                         [469]
3692 LOCATE 1,3:PRINT CHR$(20)                       [1272]
3693 IF backupflag%=true% THEN backupflag%          [4217]
=false%:LOCATE 1,4:PRINT CHR$(20):GOTO
 3696
3694 wipeflag%=true%:GOSUB 2900                      [1822]
3696 bufadr=sadr+50                                  [1661]
3697 GOSUB 2700                                      [911]
3698 GOSUB 2850                                      [905]
3699 RETURN                                          [555]
3700 REM "--- Sortieren ---"                         [2305]
3702 brokenflag%=false%                              [1464]
3705 swapflag%=false%                                [1108]
3710 FOR i=1 TO hdir-1                               [1214]
3715   IF MID$(hdir$(i),beg,l)>MID$(hdir$(          [5676]
i+1),beg,l) THEN help$=hdir$(i):hdir$
   (i)=hdir$(i+1):hdir$(i+1)=help$:sw pfla
g%=true%
3720 NEXT i                                          [375]
3725 IF swapflag%=true% THEN 3705                    [1653]
3799 RETURN                                          [555]
3800 REM "--- Belegte Bloecke Anzeigen ---          [3568]
"
3801 IF packflag%=true% THEN 3805                    [1471]
3802 GOSUB 6200:CLS#4                                [1755]
3804 GOSUB 2800                                      [907]
3805 GOSUB 2600                                      [851]
3810 FOR i=1 TO 10                                   [315]
3815   LOCATE 17,i+2:PRINT STRING$(18,143)          [1833]
;
3820 NEXT i                                          [375]
3822 LOCATE 35-offset%(format%),12:PRINT S          [4115]
PACE$(offset%(format%));
3823 blanz%=0                                        [281]
3825 FOR i=1 TO hdir                                 [1096]
3827 IF LEFT$(hdir$(i),1)>CHR$(15) THEN 38          [1623]
90
3830   FOR o=17 TO 32                                [557]
3835     posi=ASC(MID$(hdir$(i),o,1)):IF p          [3060]
osi=0 THEN 3890
3837     posi=posi-2                                 [813]
3840     LOCATE 17+posi MOD 18,posi\18+3:P          [3647]
RINT CHR$(24);CHR$(230);CHR$(24);
3842     blanz%=blanz%+1                             [500]
3845   NEXT o                                        [369]
3890 NEXT i                                          [375]
3891 IF packflag%=true% THEN packflag%=fal          [3873]
se%:GOTO 3899
3892 LOCATE 17,14:PRINT USING u2$;blanz%;:          [5256]
PRINT " Block(s) in Use"
3894 GOSUB 2850                                      [905]
3895 LOCATE 17,17:PRINT CHR$(24);"  Press           [2853]
```

ANWENDUNG

```
any Key   ";CHR$(24);
3897 GOSUB 4900:WHILE INKEY$="":WEND          [1441]
3899 RETURN                                   [555]
4100 REM "--- Ausdruck Der File - Informat    [3039]
ionen ---"
4106 printflag%=true%:pageflag%=false%:ccf    [2769]
lag%=false%
4107 GOSUB 2502                               [853]
4109 PRINT#8                                  [343]
4110 PRINT#8,STR$(total);" Byte(s) used in    [5735]
";STR$(diranz);" File(s),";
4111 PRINT#8,STR$(hidfile%);" File(s) Hidd    [4725]
en,";
4112 PRINT#8,STR$(ronfile%);" File(s) Read    [3451]
 Only."
4113 PRINT#8:PRINT#8:PRINT#8:PRINT#8:PRINT    [3109]
#8:PRINT#8:PRINT#8
4114 RETURN                                   [555]
4115 'Ueberschrift bei Druck                  [2035]
4119 PRINT#8,pinit$                           [1426]
4120 PRINT #8,CHR$(27);CHR$(45);CHR$(1);CH    [2311]
R$(124);" ";
4125 PRINT#8,"Name         ";CHR$(124);" ";   [2539]
4130 PRINT#8,"Extension    ";CHR$(124);" ";   [2146]
4135 PRINT#8,"Type         ";CHR$(124);" ";   [1754]
4140 PRINT#8,"Start        ";CHR$(124);" ";   [2323]
4145 PRINT#8,"Length       ";CHR$(124);" ";   [2490]
4150 PRINT#8,"Call         ";CHR$(124);" ";   [1883]
4155 PRINT#8,"User         ";CHR$(124);" ";   [2236]
4160 PRINT#8,"Hidden       ";CHR$(124);" ";   [2109]
4165 PRINT#8,"Read Only    ";CHR$(124);" ";   [2622]
4170 PRINT#8,CHR$(27);CHR$(45);CHR$(0);"      [2579]
";
4177 PRINT#8                                  [343]
4199 RETURN                                   [555]
4200 REM "--- Filename Aufbereiten ---"       [1782]
4205 MID$(hname$, 9,1)=CHR$(ASC(MID$(hname    [2671]
$, 9,1)) AND 127)
4210 MID$(hname$,10,1)=CHR$(ASC(MID$(hname    [2966]
$,10,1)) AND 127)
4215 hname$=LEFT$(hname$,8)+"."+MID$(hname    [2754]
$,9,3)
4299 RETURN                                   [555]
4300 REM "--- Track / Sektor Gueltig ? ---    [3865]
"
4305 IF sektor%>maxsek%(format%) THEN sekt    [3832]
or%=1:track%=track%+1
4310 IF track%>39 THEN track%=39              [1721]
4315 IF sektor%<1 THEN sektor%=maxsek%(for    [3927]
mat%):track%=track%-1
4320 IF track%<0  THEN track%=0               [2206]
4399 RETURN                                   [555]
4400 REM "--- Druck Info ---"                 [2036]
4402 GOSUB 2800                               [907]
4405 PRINT#8,CHR$(124);" ";FNsp$(12,name$)    [2299]
;
4410 PRINT#8,CHR$(124);" ";FNsp$(12,extent    [2065]
$);
4415 PRINT#8,CHR$(124);" ";FNsp$(12,type$)    [2555]
;
4420 PRINT#8,CHR$(124);     FNsp$(13,STR$(s   [2492]
tart));
4425 PRINT#8,CHR$(124);     FNsp$(13,STR$(l   [3238]
ength));
4430 PRINT#8,CHR$(124);     FNsp$(13,STR$(c   [2609]
al));
4432 PRINT#8,CHR$(124);     FNsp$(13,STR$(u   [3504]
ser%));
4435 PRINT#8,CHR$(124);" ";FNsp$(12,state$    [3156]
(hidden%));
4440 PRINT#8,CHR$(124);" ";FNsp$(12,state$    [3199]
(readon%));
4445 PRINT#8,CHR$(124)                        [1221]
4490 GOSUB 2850                               [905]
4499 RETURN                                   [555]
4500 REM "--- Rechtsbuendiege Leerzeichen     [2116]
Eliminieren ---"
4502 hflag%=false%                            [1573]
4505 FOR i=LEN(help$) TO 1 STEP -1            [1538]
4510     mi$=MID$(help$,i,1)                  [207]
4515     IF mi$=CHR$(32) AND  hflag%=false    [3228]
% THEN help$=LEFT$(help$,i-1)
4520     IF mi$<>CHR$(32) THEN hflag%=true%   [2495]
4525 NEXT i                                   [375]
4599 RETURN                                   [555]
4600 REM "--- Disk - Backup ---"              [1316]
4601 GOSUB 3250                               [1174]
4602 SPEED WRITE 1                            [1302]
4603 PRINT:PRINT TAB(5);CHR$(24);" Drive "    [4055]
;CHR$(24);" :"
4605 PRINT:PRINT TAB(5);CHR$(24);" Tape -     [3972]
Name ";CHR$(24);" :";
4606 stat%=anumeric%:xh%=1:x%=14:y%=5:ml%=    [8363]
1:eg$=drive$:GOSUB 6100:drive$=UPPER$
(eg$):GOSUB 2200:IF errflag%=true% THEN P
RINT CHR$(7);:GOTO 4606
4610 stat%=anumeric%:xh%=1:x%=20:y%=7:ml%=    [3201]
12:eg$="":GOSUB 6100
4620 PRINT:PRINT:PRINT TAB(2);CHR$(24);       [2501]
4621 GOSUB 4900:CLS#4                         [1255]
4622 PRINT " Press PLAY and REC on Tape, '    [5768]
ENTER' when done ";CHR$(24);" ";CHR$
(143)
4623 WHILE INKEY$<>CHR$(13):WEND              [1281]
4625 PRINT CHR$(11);CHR$(18):CLS#4            [2046]
4626 LOCATE 1,3::PRINT CHR$(20):backupflag    [4503]
%=true%:GOSUB 3600:GOSUB 2800:GOSUB
 3400:ziel%=ziel%*2
4627 |TAPE.OUT:|DISC.IN:CLOSEOUT:POKE modu    [3024]
s,readbl%
4628 eg$="!"+eg$+".DBK"                       [716]
4629 OPENOUT eg$:PRINT#9,format%:PRINT#9,F    [4523]
Ntrack%(ziel%):CLOSEOUT
4631 FOR track%=0 TO FNtrack%(ziel%)          [1899]
4635    FOR sektor%=1 TO 5                    [1754]
4637      bufadr=hbuf+(sektor%-1)*512         [1763]
4640      GOSUB 7000                          [867]
4645    NEXT sektor%                          [941]
4650    SAVE eg$,b,hbuf,2560                  [1312]
4655    FOR sektor%=6 TO maxsek%(format%)     [2448]
4657      bufadr=hbuf+(sektor%-6)*512         [1986]
4659      GOSUB 7000                          [867]
4661    NEXT sektor%                          [941]
4663    SAVE eg$,b,hbuf,2048                  [1088]
4665 NEXT track%                              [378]
4675 |DISC:SPEED WRITE 0                      [2845]
4680 GOSUB 2850                               [905]
4699 RETURN                                   [555]
4700 REM "--- Backup zurueck auf Disk ---"    [1972]
4701 GOSUB 6200:CLS#4                         [1755]
4702 GOSUB 4900                               [950]
4704 LOCATE 5,8:PRINT CHR$(24);" Press PLA    [6638]
Y on Tape, 'ENTER' when done ! ";
 CHR$(24);" ";CHR$(143);:LOCATE 5,8
4705 WHILE INKEY$<>CHR$(13):WEND              [1281]
4706 PRINT CHR$(20)                           [972]
4707 GOSUB 2800:GOSUB 3400                    [1191]
4708 |TAPE.IN:|DISC.OUT:CLOSEIN:POKE modus    [4474]
,writebl%
4709 OPENIN "!":INPUT#9,dbkformat%:INPUT#9    [4420]
,ziel%:CLOSEIN
4710 IF dbkformat%<>format% THEN PRINT TAB    [12686]
(1);CHR$(7);"BackUp-Disk was ";CHR$
 (24);" ";format$(dbkformat%);" ";CHR$(24)
;", but inserted Disk is ";CHR$     (24
);" ";format$(format%);" ";CHR$(24):FOR i=
1 TO 5000:NEXT i:GOTO 4799
4712 FOR track%=0 TO ziel%                    [1064]
4713    LOAD "!",hbuf                         [663]
4715    FOR sektor%=1 TO 5                    [1754]
4720      bufadr=hbuf+(sektor%-1)*512         [1763]
4725      GOSUB 7000                          [867]
4727    NEXT sektor%                          [941]
4730    LOAD "!",hbuf                         [663]
4732    FOR sektor%=6 TO maxsek%(format%)     [2448]
4734      bufadr=hbuf+(sektor%-6)*512         [1986]
4736      GOSUB 7000                          [867]
4738    NEXT sektor%                          [941]
4739 NEXT track%                              [378]
4740 |DISC                                    [839]
4750 GOSUB 2850                               [905]
4799 RETURN                                   [555]
4800 REM "--- Abbruch - Meldung ---"          [2203]
4802 brokenflag%=true%                        [1901]
4803 LOCATE 19,5:PRINT SPACE$(15);            [1910]
4805 LOCATE 19,6:PRINT CHR$(7);CHR$(24);"     [4067]
User - Break ";CHR$(24)
4806 LOCATE 19,7:PRINT SPACE$(15);            [1593]
4810 FOR i=1 TO 1000:NEXT i                   [674]
4899 RETURN                                   [555]
4900 REM "--- Input - Buffer Loeschen ---"    [3340]
4905 WHILE INKEY$<>"":WEND                    [1786]
4999 RETURN                                   [555]
5000 REM "--- Buffer auslesen ---"            [2015]
5005 GOSUB 3400                               [1009]
5010 FOR spalte%=0 TO 31                      [555]
5015    hexc$="":charc$=""                    [1712]
5020    FOR zeile%=0 TO 15                    [943]
```

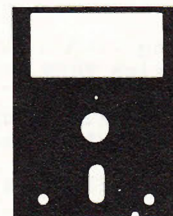
ANWENDUNG

```
5025      co%=PEEK(bufadr+zeile%+spalte%*16  [1665]
)
5030      hexc$=hexc$+HEX$(co%,2)            [1243]
5040      code$=".":IF co%>31 THEN code$=CH  [1827]
R$(co%)
5050      charc$=charc$+code$                [333]
5060    NEXT zeile%                          [571]
5070    hc$(spalte%)=hexc$:ch$(spalte%)=cha  [1583]
rc$
5090 NEXT spalte%                            [448]
5099 RETURN                                  [555]
5500 REM "--- Buffer anzeigen ---"           [2314]
5502 GOSUB 2800                              [907]
5505 IF page%<0 OR page%>1 THEN GOSUB 5800   [1575]
5520 PRINT CHR$(30);                         [994]
5525 FOR i=page%*16 TO page%*16+15           [1508]
5527   PRINT hc$(i);"  ";ch$(i)              [949]
5530 NEXT i                                  [375]
5535 GOSUB 2850                              [905]
5599 RETURN                                  [555]
5600 REM "--- Buffer editieren ---"          [2721]
5602 GOSUB 3100:GOSUB 3300:GOSUB 4900        [2874]
5605 xcur%=1:ycur%=1:GOSUB 5500              [1292]
5610 tast$=INKEY$:xold%=xcur%:yold%=ycur%    [2817]
5612 IF tast$="" AND status%=anumeric% THE   [2417]
N 5666
5614 IF tast$="" AND status%=hexa%      THE  [2915]
N 5655
5616 IF tast$=ubreak$ THEN 5699              [1272]
5618 IF tast$>CHR$(31) AND tast$<=CHR$(122   [5949]
) AND status%=anumeric% THEN GOSUB
 5700
5621 IF INSTR("0123456789ABCDEF",UPPER$(ta   [2737]
st$)) AND status%=hexa% THEN GOSUB
 5700
5622 IF tast$=CHR$( 16) THEN GOSUB 7100:GO   [1341]
TO 5610
5623 IF tast$=CHR$(  1) THEN status%=anume   [2656]
ric%:GOSUB 5500:GOSUB 3300
5624 IF tast$=CHR$(  8) THEN status%=     h  [3134]
exa%:GOSUB 5500:GOSUB 3300
5625 IF tast$=CHR$(242) THEN xcur%=xcur%-1   [2573]
5627 IF tast$=CHR$(243) THEN xcur%=xcur%+1   [2500]
5630 IF tast$=CHR$(249) THEN page%=page%+1   [6517]
:xcur%=1:ycur%=1:GOSUB 5500:GOSUB 5900
5631 IF tast$=CHR$(248) THEN page%=page%-1   [5642]
:xcur%=1:ycur%=1:GOSUB 5500:GOSUB 5900
5632 IF tast$=CHR$(241) THEN ycur%=ycur%+1   [3200]
5634 IF tast$=CHR$(240) THEN ycur%=ycur%-1   [2228]
5640 IF xcur%< 1 THEN xcur%=xcur%(status%)   [3341]
:ycur%=ycur%-1
5642 IF xcur%>xcur%(status%) THEN xcur%= 1   [2923]
:ycur%=ycur%+1
5644 IF ycur%< 1 THEN ycur%= 1               [1586]
5645 IF ycur%>16 THEN ycur%=16               [2119]
5646 IF status%=anumeric% THEN 5660          [3002]
5648 '                                       [117]
5650 LOCATE xold%,yold%:PRINT MID$(hc$(yol   [4528]
d%-1+page%*16),xold%,1)
5652 LOCATE (xold%+1)\2+34,yold%:PRINT MID   [4955]
$(ch$(yold%-1+page%*16),(xold%+1)\2,1)
5655 LOCATE xcur%,ycur%:PRINT CHR$(24);MID   [5053]
$(hc$(ycur%-1+page%*16),xcur%,1);CHR$
 (24);
5657 LOCATE (xcur%+1)\2+34,ycur%:PRINT CHR   [4978]
$(24);MID$(ch$(ycur%-1+page%*16),
 (xcur%+1)\2,1);CHR$(24);
5659 GOTO 5695                               [384]
5660 '                                       [117]
5662 LOCATE xold%+34,yold%: PRINT MID$(ch$   [2912]
(yold%-1+page%*16),xold%,1)
5664 LOCATE xold%*2-1,yold%:PRINT MID$(hc$   [4089]
(yold%-1+page%*16),xold%*2-1,2)
5666 LOCATE xcur%+34,ycur%: PRINT CHR$(24)   [4449]
;MID$(ch$(ycur%-1+page%*16),xcur%,1);
 CHR$(24)
5668 LOCATE xcur%*2-1,ycur%:PRINT CHR$(24)   [4243]
;MID$(hc$(ycur%-1+page%*16),xcur%*2-1,
 2);CHR$(24)
5695 GOTO 5610                               [448]
5699 RETURN                                  [555]
5700 REM "--- Aenderung ---"                 [1433]
5702 IF status%=anumeric% THEN 5750          [2992]
5703 tast$=UPPER$(tast$)                     [1447]
5705 LOCATE xcur%,ycur%:PRINT tast$;         [1597]
5710 MID$(hc$(ycur%-1+page%*16),xcur%,1)=t   [1879]
ast$
5715 xhelp=xcur%:IF xcur%/2-xcur%\2=0   THE  [2548]
N xhelp=xcur%-1
5720 ch$=CHR$(VAL("&"+MID$(hc$(ycur%-1+pag   [2915]
e%*16),xhelp,2)))
5725 IF ch$<CHR$(32) THEN ch$="."            [694]
5730 LOCATE (xcur%+1)\2+34,ycur%:PRINT ch$   [2106]
;
5740 MID$(ch$(ycur%-1+page%*16),(xcur%+1)\   [2854]
2,1)=ch$
5745 xcur%=xcur%+1                           [1146]
5749 GOTO 5799                               [400]
5750 '                                       [117]
5752 LOCATE xcur%+34,ycur%:PRINT tast$;      [2593]
5754 MID$(ch$(ycur%-1+page%*16),xcur%,1)=t   [2525]
ast$
5756 LOCATE xcur%*2-1,ycur%:PRINT HEX$(ASC   [2936]
(tast$),2);
5758 MID$(hc$(ycur%-1+page%*16),xcur%*2-1,   [3976]
2)=HEX$(ASC(tast$),2)
5760 xcur%=xcur%+1                           [1146]
5799 RETURN                                  [555]
5800 REM "--- Blaettern ---"                 [1338]
5805 IF page%>1 THEN 5850                    [1137]
5807 '                                       [117]
5809 page%=1                                 [709]
5810 sektor%=sektor%-1:IF sektor%<1 THEN s   [8908]
ektor%=maxsek%(format%):track%=track%
 -1:IF track%<0 THEN track%=0:sektor%=1:GO
TO 5899
5820 GOTO 5880                               [321]
5850 '                                       [117]
5853 page%=0                                 [710]
5855 sektor%=sektor%+1:IF sektor%>maxsek%(   [7600]
format%) THEN sektor%=1:track%=track%
 +1:IF track%>39 THEN track%=39:sektor%=ma
xsek%(format%):GOTO 5899
5880 POKE modus,readbl%:GOSUB 7000:GOSUB 5   [3396]
000:readflag%=true%
5899 RETURN                                  [555]
5900 REM "--- Statuszeile ---"               [1787]
5905 LOCATE#2,1,1                            [652]
5910 PRINT#2," Drive : ";drive$;"      Tra   [4068]
ck : ";USING u2$;track%;
5915 PRINT#2,"     Sector : ";USING u2$;s    [2362]
ektor%;
5920 PRINT#2,"      Page : ";USING u1$;pag   [2322]
e%+1;
5925 PRINT#2,"     Format : ";format$(forma  [2676]
t%);
5999 RETURN                                  [555]
6000 REM "--- Eingabe von Drive, Track, Se   [2922]
ktor ---"
6005 PRINT TAB(5);CHR$(24);" Drive   ";CHR$  [2102]
(24);" :    ";drive$
6007 PRINT                                   [361]
6010 PRINT TAB(5);CHR$(24);" Track   ";CHR$  [2636]
(24);" :   ";USING u1$;track%
6012 PRINT                                   [361]
6015 PRINT TAB(5);CHR$(24);" Sector  ";CHR$  [3022]
(24);" :   ";USING u1$;sektor%
6040 xh%=1:x%=17:y%=4:ml%=1:eg$=drive$:sta   [9972]
t%=anumeric%:GOSUB 6100:drive$=U PER$
 (eg$):GOSUB 2200:IF errflag%=true% THEN P
RINT CHR$(7);:GOTO 6040
6045 xh%=0:x%=17:y%=6:ml%=2:eg$=STR$(track   [8790]
%):stat%=numeric%:GOSUB 6100:track%=
VAL(eg$):IF track%<0 OR track%>39 THEN PRI
NT CHR$(7);:GOTO 6045
6050 xh%=1:x%=17:y%=8:ml%=1:eg$=STR$(sekto   [9901]
r%):stat%=numeric%:GOSUB 6100:sektor%=
 VAL(eg$):IF sektor%<1 OR sektor%>maxsek%(
format%) THEN PRINT CHR$(7);:GOTO       605
0
6060 GOSUB 2200                              [867]
6099 RETURN                                  [555]
6100 REM "--- Eingaberoutine ---"            [1687]
6101 pageflag%=false%:GOSUB 4900             [1320]
6102 IF stat%=numeric% THEN eg$=RIGHT$(eg$   [8350]
,LEN(eg$)-1):eg$=STRING$(ml%-LEN(eg$),
 CHR$(32))+eg$:heg$=SPACE$(LEN(eg$))
6103 IF stat%=anumeric% THEN eg$=eg$+SPACE   [1684]
$(ml%-LEN(eg$)):heg$=eg$
6104 xc%=1:lr$=CHR$(32):tabflag%=false%      [2753]
6105 IF xc%>ml% THEN xc%=ml%                 [2143]
6106 IF xc%<1   THEN xc%=1                   [1067]
6107 tast$=INKEY$:IF tast$="" THEN 6180      [2948]
6108 IF tast$=CHR$(13) AND stat%= numeric%   [5687]
 THEN LOCATE x%,y%:PRINT USING LEFT$
 (us$,xh%+ml%);VAL(eg$);:GOTO 6199
6109 IF tast$=CHR$(13) AND stat%=anumeric%   [3088]
```

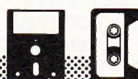
```
THEN LOCATE x%+xh%,y%:PRINT eg$;:GOTO
 6199
6110 IF tast$=CHR$(248) OR tast$=CHR$(249) [4266]
 THEN pageflag%=true%:GOTO 6199
6111 IF tast$=CHR$(16) AND xc%>=2 THEN xc% [8013]
=xc%-1:help=LEN(eg$):eg$=LEFT$(eg$,xc%
 -1):eg$=eg$+SPACE$(help-LEN(eg$ ):GOTO 61

80
6112 IF tast$=ubreak$  AND nobreak%=false [5442]
% THEN GOSUB 4800:GOTO 6199
6118 IF stat%=numeric%  AND (tast$<"0" OR  [3571]
tast$>"9")                THEN 6105
6119 IF stat%=anumeric% AND (tast$<CHR$(32 [4679]
) AND tast$>CHR$(127)) THEN 6105
6120 IF tabflag%=false% THEN eg$=heg$:tabf [2855]
lag%=true%
6125 MID$(eg$,xc%,1)=tast$                 [1129]
6130 xc%=xc%+1:cflag%=true%                [1764]
6180 LOCATE x%+xh%,y%:PRINT CHR$(24);eg$;C [3785]
HR$(24);
6185 GOTO 6105                             [402]
6199 RETURN                                [555]
6200 REM "--- Eingabe Drive ---"           [1758]
6202 GOSUB 3250                            [1174]
6205 PRINT TAB(5);CHR$(24);" Drive  ";CHR$ [2354]
(24);" : ";drive$
6210 xh%=1:x%=15:y%=4:ml%=1:eg$=drive$:sta [8350]
t%=anumeric%:GOSUB 6100:drive$=UPPER$
 (eg$):GOSUB 2200:IF errflag%=true% THEN P
RINT CHR$(7);:GOTO 6210
6215 PRINT CHR$(11);CHR$(20)               [1565]
6225 GOSUB 2200                            [867]
6299 RETURN                                [555]
7000 REM "--- Sektor lesen / schreiben --- [2360]
"

7006 GOSUB 5900                            [1053]
7008 POKE sadr+14,drive%                   [837]
7010 POKE sadr+16,track%                   [319]
7015 POKE sadr+18,sektor%-1+form%          [1093]
7020 POKE sadr+20,bufadr-256*INT(bufadr/25 [1650]
6)
7025 POKE sadr+21,INT(bufadr/256)          [1293]
7030 CALL sadr                             [85]
7099 RETURN                                [555]
7100 REM "--- Druck ---"                   [1645]
7102 PRINT#8,pinit$                        [1426]
7103 PRINT#8,"Drive : ";drive$;"     Track [4922]
 : ";USING u1$;track%;
7104 PRINT#8,"      Sektor : ";USING u1$;se [3033]
ktor%:PRINT#8
7105 FOR i=0 TO 31 STEP 2                  [1244]
7107    FOR p=1 TO 8                       [624]
7109       PRINT#8,MID$(hc$(i)+hc$(i+1),(p-1 [2031]
)*8+1,8);" ";
7111    NEXT p                             [364]
7112    PRINT #8,"      ";                 [628]
7115    FOR o=1 TO 64                      [925]
7120       help$=MID$(ch$(i)+ch$(i+1),o,1):I [2843]
F help$>CHR$(127) THEN help$="."
7125       PRINT#8,help$;                  [1054]
7130    NEXT o                             [369]
7135    PRINT#8                            [343]
7150 NEXT i                                [375]
7155 PRINT#8:PRINT#8:PRINT#8:PRINT#8:PRINT [3852]
#8:PRINT#8:PRINT#8:PRINT#8
7199 RETURN                                [555]
8000 REM "--- Daten laden ---"             [2424]
8005 sadr=42024:bufadr=sadr+50:hbuf=sadr-2 [3908]
561:MEMORY hbuf-1:GOSUB 10000
8007 DEF FNtrack%(block)=INT(dir%(format%) [4177]
+block/maxsek%(format%))
8008 DEF FNsektor%(block)=dir%(format%)*ma [7384]
xsek%(format%)-maxsek%(format%)*(INT(
 dir%(format%)+block/maxsek%(format%)))+bl
ock+1
8009 DEF FNsp$(ml%,var$)=var$+SPACE$(ml%-L [3206]
EN(var$))
8010 DIM format$(3),ziel%(3),offset%(3),hc [10631]
$(31),ch$(31),track%(3),maxsek(3),cur$
 (12),xcur%(3),hdir%(64),dir%(3),type$(3),
htrack%(4),hsektor%(4),help$(5),     ide
nt(1),state$(1)
8013 sure$=CHR$(7)+CHR$(24)+" Are You Sure [3562]
 ? (Y/N) "+CHR$(24)
8014 ident(0)=&A89F:ident(1)=&A8DF         [1597]
8015 state$(0)=" ":state$(1)="*"           [1504]
8021 drive$="A":sektor%=1:format%=1        [1195]
8022 readbl%=&84:writebl%=&85:modus=sadr+2 [2367]
6
8023 numeric%=1:anumeric%=2:hexa%=3:xcur%( [5445]
anumeric%)=16:xcur%(hexa%)=32:status%=
 anumeric%
8024 true%=1:false%=0:readflag%=false%:che [8131]
ckflag%=false%:nobreak%=true%:errflag%
 =false%
8025 ver$="3.0"                            [968]
8026 us$="######":u0$="#####":u1$="##":u2$ [810]
="###"
8027 ubreak$=CHR$(127)                     [1283]
8028 active=TIME                           [925]
8030 pinit$=CHR$(27)+CHR$(120)+CHR$(0)+CHR [3486]
$(27)+CHR$(55)+CHR$(15)
8031 message$="Good-Bye from "+CHR$(24)+"  [5756]
Dis-K-Patcher "+CHR$(24)+" Ver "+ver$
8032 RESTORE 8040                          [703]
8033 FOR i=1 TO 3:READ h$:format$(i)=FNsp$ [2416]
(4,h$):NEXT i
8034 FOR i=0 TO 3:READ h$:type$(i)=FNsp$(1 [2954]
1,h$):NEXT i
8035 FOR i=1 TO 3:READ maxsek%(i):NEXT i   [1316]
8036 FOR i=1 TO 3:READ dir%(i):NEXT i      [1610]
8037 FOR i=1 TO 3:READ ziel%(i):NEXT i     [1688]
8038 FOR i=1 TO 3:READ offset%(i):NEXT i   [1222]
8039 FOR i=1 TO 12:READ cur$(i):NEXT i     [1747]
8040 DATA CP/M,Data,IBM                    [1075]
8041 DATA Basic,Basic Prot.,Binary,ASCII   [2617]
8042 DATA 9,9,8                            [186]
8043 DATA 2,0,1                            [180]
8044 DATA 170,179,155                      [831]
8045 DATA 11,2,7                           [386]
8049 DATA Read Sector,Write Sector,Searchi [4274]
ng For Data,Wipe Disk
8050 DATA UnErase Files,Information,Sort D [4744]
ir,Pack Disk,Block Partition
8051 DATA Print Information,Backup,Restore [2312]
8059 INK 0,13:INK 1,0:BORDER 13:PAPER 0:PE [2973]
N 1:MODE 2
8060 RESTORE 8100                          [914]
8065 FOR wnum%=0 TO 3:READ xl%,xr%,yo%,yu% [2567]
8070 LOCATE xl%-1,yo%-1:PRINT CHR$(150);ST [4931]
RING$(xr%-xl%+1,CHR$(154));CHR$(156);
8072 FOR i=yo% TO yu%:LOCATE xl%-1,i:PRINT [2627]
 CHR$(149);:LOCATE xr%+1,i:PRINT CHR$
 (149);:NEXT i
8074 LOCATE xl%-1,yu%+1:PRINT CHR$(147);ST [3371]
RING$(xr%-xl%+1,CHR$(154));CHR$(153);
8075 NEXT wnum%                            [416]
8080 RESTORE 8100:FOR wnum%=0 TO 3:READ xl [6312]
%,xr%,yo%,yu%:WINDOW#wnum%,xl%,xr%,yo%
 ,yu%:NEXT wnum%
8085 WINDOW#4,55,79,9,21                   [1415]
8090 CLS#1:PRINT#1,TAB(20);"Dis-K-Patcher  [7118]
 Ver ";ver$;"    ";CHR$(164);
 "  1986 by ArtWare  GbR";
8094 PAPER#3,1:PEN#3,0:CLS#3:PAPER#4,1:PEN [2375]
#4,0:CLS#4
8095 CLS#3:PRINT#3:PRINT#3,TAB(9);" Editor [2798]
":PRINT#3,TAB(9);STRING$(8,154)
8097 GOSUB 5900                            [1053]
8098 trash=FRE("")                         [595]
8099 RETURN                                [555]
8100 '                                     [117]
8105 DATA 2,52, 5,21                       [504]
8110 DATA 2,79, 2, 2                       [710]
8115 DATA 2,79,24,24                       [733]
8120 DATA 55,79,5,21                       [504]
8199 RETURN                                [555]
10000 REM                                  [272]
10005 sum%=0:RESTORE 10015:FOR i=sadr TO s [4708]
adr+25:READ help$:help=VAL("&"+help$):
  sum%=sum%+help:POKE i,help:NEXT i
10010 IF sum%<>&9CF THEN MODE 2:PRINT CHR$ [4335]
(7);"Error in Data...":END
10015 DATA 21,42,A4,CD,D4,BC,22,43,A4,79,3 [4627]
2,45,A4,1E,00,16,00,0E,00,21,40,9C,DF,
 43,A4,C9
10099 RETURN                               [555]
```



Tabwrite



Mit dem Programm »Tabwrite« ist es möglich, Anlagen, die eine Spalte zur Addition von DM-Beträgen enthalten, zu erstellen. Eine Anlagezeile und zwei Textzeilen stehen dem User zur Verfügung. Eine Textendzeile erlaubt noch diverse Mitteilungen.

Beispiele:
Einkommen- bzw. Lohnsteueranlagen (Jahresausgleich), Tabellen für Versicherungen, Leistungsabrechnungen für Krankenversicherungen (Privatpatienten), sonstige Tabellen. Selbst Mahnungen sind möglich.
Das Programm ist für Diskette und Kassette geeignet. Bei der Diskettenversion muß lediglich Zeile 2050 und 3050 entfernt werden (CAT-Anweisung).

Auf der Grundlage des Tabwriters ist es auch möglich, ein kleines Faktura-Programm zu erstellen, mit Artikelbezeichnungen, die mit der Artikelnummer aufgerufen werden könnten. Ein Druckerspooler sorgt für schnellen Ausdruck, so daß die nächste Rechnung schon während des Druckens wieder erfaßt werden könnte.

Variablenliste:

a und t	Laufvariablen
n	Schleifen
	Verlassen der Eingabe (Klammeraffe)
anzahl	Höchstzahl (Maximum)
dat$	Datum
anr$	Anrede
name$	Vorname Name
stras$	Straße
ort$	PLZ/Ort
anlage$	Anlagebezeichnung
text1$	Kurzmitteilung an Empfänger 1. Zeile
text2$	Kurzmitteilung an Empfänger 2. Zeile
text3$	Eingabe einer Schlußzeile
bel$	Belegbezeichnung Nr/Buchstabe
text$	Bezeichnung des Beleges
dm	Betrag in DM
zwisu	Zwischensumme
esu	Endsummer
z$(1) - z$(3)	Eingabe des Absenders

Hinweise zum Eintippen:
Wer kein Diskettenlaufwerk besitzt, muß die Option Laden/Sichern abändern.
In den Zeilen 9300 – 9330 muß die Anschrift des Users eingegeben werden. Die Druckersteuerzeichen müssen eventuell auf den, vom User benutzten, Drucker umgestellt werden. (Das Programm ist auf Schneider NLQ 401 eingestellt.)

Mit run"Tabwrite" gestartet, erscheint das Hauptmenue. Mit der Option 6 erhalten Sie die Benutzerinformation. Ansonsten ist das Programm ausreichend dokumentiert, so daß es keine Schwierigkeiten geben dürfte.

Hans Müller

ANWENDUNG

ANWENDUNG

```
5 REM *** T
abwrite Diskettenversion ***        [2050]
10 MODE 1:OPENOUT "D":MEMORY HIMEM-1:CLOSE [3647]
OUT
20 WINDOW#1,1,40,1,3:WINDOW#2,1,40,4,23:WI [3777]
NDOW #3,1,40,23,25
30 INK 0,13:INK 1,0:BORDER 9              [1717]
40 PAPER#1,2:PEN#1,3:PAPER#2,0:PEN#2,1:PAP [4245]
ER#3,2:PEN#3,3
50 CLS:CLS#1:CLS#2:CLS#3                  [118]
60 PRINT #1:PRINT #1,"#################### [2516]
####################"
70 LOCATE#2,5,4:PRINT #2,"** CPC 464-T A B [3316]
 W R I T E R **"
80 LOCATE#2,8,8:PRINT#2,"Programm zum Erst [6743]
ellen von ":LOCATE#2,15,10:PRINT#2," Anlag
en!"
90 LOCATE#2,9,15: PRINT #2," @ 1986 by Han [1847]
s Mueller"
100 PRINT#3:PRINT#3,"#################### [2226]
###################"
120 FOR a=1 TO 1500:NEXT                  [1251]
140 REM *** Umschaltung mode 2 Windows / F [2147]
arben neu setzen  ***
150 MODE 2:BORDER 13                      [1309]
160 WINDOW#1,1,80,1,2:                    [1011]
170 WINDOW#2,1,80,4,23                    [1415]
180 WINDOW#3,1,80,24,25                   [1385]
190 PAPER#1,0:PEN#1,13:PAPER#2,13:PEN#2,0: [5505]
PAPER#3,0:PEN#3,13
200 DIM text$(60),dm(60),bel$(60),bel(60)  [2771]
210 REM                                   [272]
220 REM *** Programmkopf ***              [1393]
230 CLS:CLS#1:CLS#2:CLS#3                 [118]
235 LOCATE#1,22,2:PRINT#1,"M E N U E   -  [2453]
 T A B W R I T E R"
240 LOCATE 27,8:PRINT" 1 = Daten EINGABE " [4781]
:LOCATE 27,10:PRINT" 2 = Anlagen laden "
250 LOCATE 27,12:PRINT" 3 = Anlagen speich [12931]
ern ":LOCATE 27,14:PRINT" 4 = Ausgabe auf
 Bildschirm ":LOCATE 27,16: PRINT" 5 = Aus
gabe auf Drucker ":LOCATE 27,18:PRINT" 6 =
 Programminformation! "
260 LOCATE 27,20:PRINT" 7 = Programm beend [3284]
en "
270 LOCATE#3,18,2:PRINT#3,"   Bitte waehle [3365]
n   <  1   bis   7  > "
280 a$=INKEY$:IF a$="" THEN 280           [1408]
290 IF a$<"1" OR a$>"7" THEN PRINT CHR$(7) [2995]
:GOTO 270
300 ON VAL(a$) GOTO 400,2000,3000,4000,600 [2315]
0,7000,8000
310 GOTO 220                              [425]
400 REM **** Untermenue DATEN - Eingabe ** [2093]
**
410 CLS:CLS#1:CLS#2:CLS#3                 [118]
420 LOCATE#1,17,2:PRINT#1," ****   Unterme [4263]
nue fuer DATEN - EINGABE  ****
430 LOCATE#2,20,6:PRINT#2," 1 = Datum und  [2016]
Adresse "
440 LOCATE#2,20,8:PRINT#2," 2 = Anlagebeze [6361]
ichnung/Betreff/Steuernr./Vers.Nr."
450 LOCATE#2,20,10:PRINT#2," 3 = Kurztext  [4684]
fuer Mitteilung ( 3 Zeilen) "
460 LOCATE#2,20,12:PRINT#2," 4 = Tastenbel [4093]
egung Zehnerblock "
470 LOCATE#2,20,14:PRINT#2," 5 = Eingabe d [3263]
er Daten Tabelle "
480 LOCATE#2,20,16:PRINT#2," 6 = Drucker I [3419]
nitialisieren "
485 LOCATE#2,20,18:PRINT#2," 7 = Zum Haupt [2841]
menue "
490 LOCATE#3,18,2:PRINT#3,"   Bitte waehle [3365]
n   <  1   bis   7  > "
500 a$=INKEY$:IF a$="" THEN 500           [1380]
510 ON VAL(a$) GOTO 600,800,700,9000,1000, [3072]
900,220
520 IF a$<"1" OR a$>"7" THEN PRINT CHR$(7) [3037]
:GOTO 500
600 REM ** Datum und Adresse **           [1962]
610 CLS:CLS#1:CLS#2:CLS#3:LOCATE#1,20,2:PR [2424]
INT#1," ****    Datum und Adresse  eingeben
  ****"
620 LOCATE#2,2,2:PRINT#2,"Datum        ==> [4265]
 ";:INPUT#2,dat$
630 LOCATE#2,2,4:PRINT#2,"Anrede       ==> [3506]
 ";:INPUT#2,anr$
640 LOCATE#2,2,6:PRINT#2,"Vorname Name ==> [3545]
 ";:INPUT#2,name$
650 LOCATE#2,2,8:PRINT#2,"Strasse Nr.  ==> [1975]
 ";:INPUT#2,stras$
660 LOCATE#2,2,10:PRINT#2,"PLZ Ort      == [3799]
> ";:INPUT#2,ort$
670 LOCATE#2,14,16:PRINT#2,"Daten  richtig [4779]
 eingegeben ?      < J/N > "
680 a$=UPPER$(INKEY$):IF a$="" THEN 680    [1844]
690 IF a$="N" THEN CLS#2: GOTO 620 ELSE 41 [1638]
0
700 REM ****  Kurztext 3 Zeilen fuer Mitte [2814]
ilung an Empfaenger *****
710 CLS:CLS#1:CLS#2:CLS#3:LOCATE#1,20,2:PR [4620]
INT#1," ****    Bis 3 Zeilen Kurzmitteilung
  ****"
720 LOCATE#2,2,2:PRINT#2,"* Kurzmitteilung [13418]
en fuer Empfaenger eingeben  (72) Zeichen
je Zeile! *":LOCATE#2,2,4:PRINT#2,"* 2 Zei
len zwischen Anschrift und Tabelle * 1 Zei
le am Ende *"
730 LOCATE#2,2,6:PRINT#2,"1) ==> ";:LINE I [3220]
NPUT#2,"",text1$
735 IF LEN(text1$(n))>72 THEN LOCATE#2,14, [8462]
18:PRINT#2,"Text zu lang.....":PRINT CHR$(
7):FOR a=1 TO 1500:NEXT a:CLS#2:GOTO 730
740 LOCATE#2,2,8:PRINT#2,"2) ==> ";:LINE I [2606]
NPUT#2,"",text2$
750 IF LEN(text2$(n))>72 THEN LOCATE#2,14, [6650]
18:PRINT#2,"Text zu lang.....":PRINT CHR$(
7):FOR a=1 TO 1500:NEXT a:CLS#2:GOTO 740
755 LOCATE#2,2,10:PRINT#2,"3) Endzeile ==> [4363]
 ";:LINE INPUT#2,"",text3$
760 IF LEN(text3$(n))>72 THEN LOCATE#2,14, [6605]
18:PRINT#2,"Text zu lang.....":PRINT CHR$(
7):FOR a=1 TO 1500:NEXT a:CLS#2:GOTO 755
770 LOCATE#2,14,16:PRINT#2,"Daten  richtig [4779]
 eingegeben ?      < J/N > "
780 a$=UPPER$(INKEY$):IF a$="" THEN 780    [1840]
790 IF a$="N" THEN CLS#2: GOTO 720 ELSE 41 [1312]
0
800 REM ** Eingabe der Anlagebezeichnung * [1814]
*
810 CLS:CLS#1:CLS#2:CLS#3:LOCATE#1,17,2:PR [4791]
INT#1," ****     Eingabe der Anlagebezeich
nung     ****"
820 LOCATE#2,2,6:INPUT#2,"",anlage$        [2055]
850 LOCATE#2,14,16:PRINT#2,"Daten  richtig [4779]
 eingegeben ?      < J/N > "
860 a$=UPPER$(INKEY$):IF a$="" THEN 860    [1840]
870 IF a$="N" THEN CLS#2: GOTO 820 ELSE 41 [1627]
0
900 REM ** Drucker Initialisieren fuer NLQ [1697]
 401 von Schneider **
910 CLS:CLS#1:CLS#2:CLS#3:LOCATE#1,20,2:PR [4546]
INT#1," ****     Drucker Initialisieren
 ****"
920 LOCATE#3,2,2:PRINT#3,"Zum Schreiben in [6446]
 Schmalschrift erst Option 2 druecken!! "
930 LOCATE#3,60,2:PRINT#3,"Mit ENTER zurue [5755]
ck...":LOCATE#2,28,18:PRINT#2,"Drucker ein
schalten!!":PRINT CHR$(7)
940 LOCATE#2,2,6:PRINT#2," Einschalten NLQ [9795]
   =  1        Ausschalten        =  2
":LOCATE#2,2,10:PRINT#2," Schmalschrift ei
n   =  3       Schmalschrift aus   =  4"
950 a$=INKEY$:IF a$="" THEN 950           [1322]
960 IF a$="1" THEN PRINT#8,CHR$(27);CHR$(1 [2913]
20);CHR$(1):GOTO 950
970 IF a$="2" THEN PRINT#8,CHR$(27);CHR$(1 [2824]
20);CHR$(0):GOTO 950
980 IF a$="3" THEN PRINT#8,CHR$(15):GOTO 9 [2915]
50
985 IF a$="4" THEN PRINT#8,CHR$(18):GOTO 9 [2397]
50
990 GOTO 410                              [450]
1000 REM ** Dateieingabe ***              [2299]
1010 CLS:CLS#1:CLS#2:CLS#3                [118]
1020 LOCATE#1,2,2:PRINT#1," Dateneingabe " [2245]
1030 zaehler=0:anzahl=0                   [1597]
1040 FOR n=1 TO 60                        [900]
1090 LOCATE#2,1,2:PRINT#2,"Dateneingabe be [5670]
enden, bei Beleg Nr. <@> eingeben!"
1100 PRINT#2                              [317]
1110 LOCATE#1,65,2:PRINT#1,"Lfd.-Nr.";n   [2797]
1120 LOCATE#2,1,6:PRINT#2,"Beleg Nr.    == [3508]
> ";:INPUT#2,"",bel$(n)
1130 IF bel$(n)="@" THEN zaehler=1:anzahl= [2948]
n-1:GOTO 1310
```

```
1150 LOCATE#2,1,4:PRINT#2,"Texteingabe bis  [2536]
 40 Zeichen !!"
1160 LOCATE#2,1,9:PRINT#2,"Text          ==  [4357]
> ";:INPUT#2,"",text$(n)
1170 IF LEN(text$(n))>40 THEN LOCATE#2,14,  [7101]
18:PRINT#2,"Text zu lang.....":PRINT CHR$(
7):FOR a=1 TO 1500:NEXT a:CLS#2:GOTO 1160
1180 LOCATE#2,1,12:PRINT#2,"Betrag DM:    =  [3936]
=> ";:INPUT#2,"",dm(n)
1200 LOCATE#2,14,16                          [938]
1210 PRINT#2,"Daten  richtig eingegeben ?   [3862]
  < J/N > "
1220 a$=UPPER$(INKEY$):IF a$="" THEN 1220   [1588]
1230 IF a$="J" THEN CLS#2:GOTO 1260         [1861]
1240 IF a$="N" THEN CLS#2:GOTO 1090         [1284]
1250 SOUND 1,200,50,7:SOUND 2,201,50,7:GOT  [3055]
O 1220
1260 zwisu=zwisu+dm(n)                      [1326]
1270 LOCATE#3,5,2:PRINT#3,"Zwischensumme D  [4396]
M : ";USING"######.##";zwisu
1280 IF n >=60 THEN anzahl=n-1:GOTO 1310     [832]
1300 NEXT n                                  [366]
1310 esu=esu+zwisu                          [1793]
1320 GOTO 220                                [425]
2000 REM *** Anlagen Laden von Diskette/Ka  [2358]
ssette ***
2010 CLS:CLS#1:CLS#2:CLS#3                   [118]
2020 LOCATE#1,27,2:PRINT#1,"Laden von Disk  [3022]
ette/Kassette"
2030 LOCATE#3,10,2:PRINT#3,"Diskette/Kasse  [6367]
tte einlegen.......";:PRINT CHR$(7):FOR a
=1 TO 1500:NEXT a
2040 GOSUB 9500                              [844]
2050 LOCATE #2,2,1:PRINT#2,"Directory der   [9430]
eingelegten Discette......":WINDOW SWAP 0,
2:LOCATE#2,2,6:CAT:WINDOW SWAP 2,0:REM ***
 Zeile bei Kassettembetrieb entfernen ***
2060 LOCATE #2,2,19:PRINT#2,"Bitte geben S  [5186]
ie den Namen ein (ohne extention!)"
2070 LOCATE#2,2,20:INPUT#2,"",nam$:nam$=na  [2252]
m$+".tab"
2080 IF LEN(nam$)>12 THEN LOCATE#2,20,18:P  [5064]
RINT#2,"Name zu lang...":CLS#2:FOR a=1 TO
1200:NEXT a:GOTO 2060
2090 OPENIN nam$                            [1020]
2100 LINE INPUT#9,dat$                      [1564]
2110 LINE INPUT#9,anr$:LINE INPUT#9,name$:  [4709]
LINE INPUT#9,stras$:LINE INPUT#9,ort$
2120 LINE INPUT#9,anlage$                   [1537]
2130 LINE INPUT#9,text1$:LINE INPUT#9,text  [6172]
2$:LINE INPUT#9,text3$
2150 INPUT#9,zwisu:INPUT#9,esu:INPUT#9,anz  [2813]
ahl
2160 FOR n=1 TO 60:zaehler=0                [1767]
2170 IF EOF THEN 2290                        [394]
2180 LINE INPUT#9,bel$(n):LINE INPUT#9,tex  [4455]
t$(n):INPUT#9,dm(n)
2190 REM                                     [272]
2220 NEXT n                                  [366]
2290 CLOSEIN                                 [752]
2300 GOTO 220                                [425]
3000 REM *** Anlagen Sichern auf Diskette/  [1668]
Kassette ***
3010 CLS:CLS#1:CLS#2:CLS#3                   [118]
3020 LOCATE#1,27,2:PRINT#1,"Sichern auf Di  [2633]
skette/Kassette"
3030 LOCATE#3,10,2:PRINT#3,"Diskette/Kasse  [6606]
tte einlegen.......";:PRINT CHR$(7):FOR a=
1 TO 1500: NEXT a
3040 GOSUB 9500                              [844]
3050 LOCATE #2,2,1:PRINT#2,"Directory der   [9430]
eingelegten Discette......":WINDOW SWAP 0,
2:LOCATE#2,2,6:CAT:WINDOW SWAP 2,0:REM ***
 Zeile bei Kassettembetrieb entfernen ***
3080 LOCATE#2,2,19:PRINT#2,"Bitte geben Si  [5186]
e den Namen ein (ohne extention!)"
3090 LOCATE#2,2,20:INPUT#2,"",nam$:nam$=na  [2252]
m$+".tab"
3100 IF LEN(nam$)>12 THEN LOCATE#2,20,18:P  [5068]
RINT#2,"Name zu lang...":CLS#2:FOR a=1 TO
1200:NEXT a:GOTO 3080
3110 OPENOUT nam$                            [773]
3120 PRINT#9,dat$                            [978]
3130 PRINT#9,anr$:PRINT#9,name$:PRINT#9,st  [2922]
ras$:PRINT#9,ort$
3140 PRINT#9,anlage$:PRINT#9,text1$:PRINT#  [5027]
9,text2$:PRINT#9,text3$
3150 PRINT#9,zwisu:PRINT#9,esu:PRINT#9,anz  [3380]
ahl
3160 FOR n=1 TO anzahl                       [833]
3170 REM                                     [272]
3180 PRINT#9,bel$(n):PRINT#9,text$(n):PRIN  [4560]
T#9,dm(n)
3190 REM                                     [272]
3200 NEXT n                                  [366]
3210 CLOSEOUT                                [902]
3220 GOTO 220                                [425]
4000 REM *** Bildschirmausgabe ****         [1646]
4010 CLS:CLS#1:CLS#2:CLS#3                   [118]
4020 LOCATE#1,23,2:PRINT#1,"  B I L D S C   [2762]
H I R M A U S G A B E "
4030 zaehler=0                               [699]
4040 PRINT#2,anr$:PRINT#2,name$:PRINT#2,st  [5463]
ras$:PRINT#2,"":PRINT#2,ort$
4050 PRINT#2,anlage$                         [454]
4060 PRINT#2,text1$                          [313]
4070 PRINT#2,text2$:PRINT#2:PRINT#2         [1860]
4080 PRINT#2,"Beleg"                         [909]
4090 PRINT#2,"Nr.";:PRINT#2,"    T E X T";  [3251]
4100 PRINT#2,TAB(63);"Summe  "              [1635]
4110 PRINT#2,STRING$(80,"-")                [1476]
4115 GOSUB 9500                              [844]
4120 FOR n=1 TO anzahl:zaehler=zaehler+1    [2696]
4130 IF text$(n)="@" THEN 4250              [1590]
4140 IF zaehler>15 THEN GOSUB 9500:zaehler  [1876]
=0
4160 PRINT#2,TAB(2);USING"\     \";bel$(n);  [2296]
4170 PRINT#2,TAB(10);USING"\                [3459]
                          \";text$(n);
4180 PRINT#2,TAB(59);USING"######.##";dm(n  [1136]
)
4230 NEXT n                                  [366]
4240 REM *** Schlussrechnung ***            [1265]
4250 PRINT#2,STRING$(80,95);                [1534]
4260 PRINT#2," Endsumme DM";:PRINT#2,TAB(5  [3556]
9);USING"######.##";esu
4270 PRINT#2                                 [317]
4280 PRINT#2,text3$                          [309]
4290 GOSUB 9500                              [844]
4300 CLS#3:PRINT#3:PRINT#3," Keine weitere  [3345]
n Daten vorhanden....."
4310 GOTO 220                                [425]
6000 REM ***  DRUCKERAUSGABE  ***           [1418]
6010 CLS:CLS#1:CLS#2:CLS#3                   [118]
6020 LOCATE#1,25,2:PRINT#1," D R U C K E R  [2643]
 A U S G A B E "
6030 LOCATE#2,28,18:PRINT#2,"Drucker einsc  [5752]
halten!!":PRINT CHR$(7)
6040 LOCATE#2,20,12:PRINT#2,"A C H T U N G  [4167]
      es wird gedruckt.....!"
6050 LOCATE#2,10,4:PRINT#2,"< 1 > =  mit    [3879]
  < 2 > = ohne  Anschrift   < 3 > = Nur Ta
belle "
6060 a$=INKEY$:IF a$="" THEN 6060           [1271]
6070 IF a$="1" THEN GOSUB 9300:GOTO 6120    [2109]
6080 IF a$="2" THEN GOSUB 9300:GOTO 6200    [1463]
6090 IF a$="3" THEN GOTO 6400               [1561]
6110 REM *** Anschrift drucken ***          [1896]
6120 PRINT#8:PRINT#8:PRINT#8:PRINT#8        [1873]
6130 PRINT#8,TAB(7);anr$                    [1595]
6140 PRINT#8,TAB(7);name$                   [1892]
6150 PRINT#8,TAB(7);stras$:PRINT#8          [2149]
6160 PRINT#8,TAB(7);ort$                    [1798]
6170 PRINT#8:PRINT#8:PRINT#8                [1827]
6200 PRINT#8,TAB(7);anlage$:PRINT#8:PRINT#  [3758]
8
6210 PRINT#8,TAB(7);text1$                  [1700]
6220 PRINT#8,TAB(7);text2$                  [1706]
6230 PRINT#8:PRINT#8                        [1314]
6300 REM                                     [272]
6380 REM                                     [272]
6400 REM  *** Ausdruck nur Tabelle ***      [1371]
6440 PRINT#8,TAB(7);"Beleg.Nr.";            [1668]
6450 PRINT#8,TAB(21);"Bezeichnung des Bele  [3012]
ges";
6460 PRINT#8,TAB(67);"Summe DM "            [1570]
6470 PRINT#8,TAB(7);STRING$(68,"_")         [2719]
6480 FOR n=1 TO anzahl                       [833]
6490 IF bel$(n)="@" THEN 6590               [1162]
6500 bel(n)=VAL(bel$(n)):PRINT#8,TAB(7);US  [2665]
ING"###";bel(n);
6520 PRINT#8,TAB(14);USING"\                [3883]
                          \";text$(n)
;
6530 PRINT#8,TAB(66);USING"######.##";dm(n  [2191]
);
6540 NEXT                                    [350]
6550 REM ** Schlussrechnung ***             [1827]
```

```
6590 PRINT#8,TAB(7);STRING$(68,"_")           [2719]
6600 PRINT#8,TAB(50);"Gesamtsumme DM ";       [2579]
6610 PRINT#8,TAB(66);USING"######.##";esu;    [1858]
6620 PRINT#8                                  [343]
6630 PRINT#8,TAB(64);STRING$(11,61)           [1639]
6640 PRINT#8:PRINT#8                          [1314]
6650 PRINT#8,TAB(7);text3$                    [1728]
6660 FOR a=1 TO 4:PRINT#8:NEXT a              [1814]
6670 CLS#3                                    [371]
6680 LOCATE#3,10,2:PRINT#3,"Keine weiteren [5748]
 DATEN mehr vorhanden......!"
6690 GOSUB 9500                               [844]
6700 GOTO 220                                 [425]
7000 REM *** Programminformationen ***        [2373]
7010 CLS:CLS#1:CLS#2:CLS#3                    [118]
7020 LOCATE#1,20,2:PRINT#1," ****    Progra [3925]
mminformationen    ****"
7030 PRINT#2:PRINT#2," TABWRITE wurde konz [4278]
ipiert zum Erstellen von Anlagen (Listen),
 die eine Spalte"
7040 PRINT#2," zum Addieren von Summen vor [8188]
sieht. Z.B.:Steuer/Versicherung/Krankenkas
se."
7050 PRINT#2," Kurzbeschreibung zum Ablauf [10202]
:":PRINT#2," Das Hauptmenue ist selbsterkl
aerend. Waehlen Sie Option 1 = Dateneingab
e! "
7060 PRINT#2," Utermenue DATEN - Eingabe: [11016]
":PRINT#2," Sie koennen mit 1 beginnend, d
ie von Ihnen gewuenschten Optionen anwaehl
en"
7070 PRINT#2," und dort die Daten Eingeben [6684]
. ":PRINT#2:PRINT#2,"  OPTION       EINGA
BE                oder: "
7080 PRINT#2,:PRINT#2,"  == 1 ==         Ein [4734]
gabe Datum              ENTER"
7090 PRINT#2,"    == 1 ==       Anschrift  [2681]
              ENTER"
7100 PRINT#2,"    == 2 ==       Bezeichnung [4734]
der Anlage     ENTER"
7110 PRINT#2,"    == 3 ==       1. Textzeile [3708]
              ENTER"
7120 PRINT#2,"    == 3 ==       2. Textzeile [2179]
              ENTER"
7130 PRINT#2,"    == 3 ==       3. Textzeile [3793]
 Endzeile     ENTER"
7140 PRINT#2,"    == 4 ==       Tastenbelegu [5686]
ng 0-9 Zehnerblock oder   ENTER "
7150 PRINT#2,"    == 5 ==       Dateneingabe [7574]
 Tabelle, wird Mit  - @ - = Klammeraffe be
i "
7160 GOSUB 9500:PRINT#2," Beleg.Nr. verlas [9479]
sen. Eine neue Tabelle kann nur ueber Prog
ramm beenden, "
7170 PRINT#2," Hauptmenue Option 7 erstell [12078]
t werden. Mit zweimal ESC-Taste und RUN !!
 ":PRINT#2," kann man ebenfalls das Progra
mm verlassen und neu beginnen."
7200 PRINT#2:PRINT#2," Noch ein Hinweis zu [8380]
m Kleindruck!! Man muss beim NLQ 401 immer
 erst die Opt. 2"
7210 PRINT#2," (NLQ aus) eingeben, um den [7782]
Kleindruck mit Option 3 aktivieren zu koen
nen!"
7220 GOSUB 9500                               [844]
7230 GOTO 220                                 [425]
8000 REM *** Programm beenden   ****          [1458]
8010 CLS:CLS#1:CLS#2:CLS#3                    [118]
8020 LOCATE#1,16,2:PRINT#1," ****     Prog [2871]
ramm   B e e n d e n      ****"
8030 LOCATE#3,18,2:PRINT#3,"Achtung!! Alle [3974]
 DATEN werden geloescht!!"
8040 LOCATE#2,15,10:PRINT#2,"Wollen Sie da [6293]
s Programm beenden      < J/N > ?"
8050 a$=UPPER$(INKEY$):IF a$="" THEN 8050: [7080]
IF a$="J" THEN LOCATE#2,20,14:PRINT#2,"All
e Daten wurden geloescht.....!":GOTO 8110
8060 IF a$="N" THEN GOTO 220                  [899]
8110 CLS#3:LOCATE#3,25,2:PRINT#3,"Tschuess [5397]
! So long und good bye.....":FOR a=1 TO 10
00:NEXT a
8120 END                                      [110]
9000 REM **** Tastenbelegung des Zehnerblo [1665]
ckes *****
9010 CLS:CLS#1:CLS#2:CLS#3:LOCATE#1,16,2:P [3715]
RINT#1," ****    Tastenbelegung des Zehnerb
lockes     ****"
9040 CLS#2:LOCATE#3,25,2::PRINT#3,"Aendern [4631]
 der Tastenbelegung......"
9050 FOR t=0 TO 9:PRINT#2,"Taste";t;":":LI [4563]
NE INPUT#2," ",k$(t):NEXT t
9060 CLS#3:LOCATE#3,20,2:PRINT#3,"Daten ri [6007]
chtig eingegeben   < J/N > ? "
9070 a$=UPPER$(INKEY$):IF a$="" THEN 9070  [3302]
ELSE IF a$="N" THEN 9040
9080 IF a$<>"J" THEN CLS#2:GOTO 9070          [1590]
9090 FOR t=0 TO 9:KEY t,k$(t):NEXT t          [1985]
9100 GOTO 410                                 [450]
9300 REM    *** Eingabe  Absender  fest ***   [1835]
9310 z$(1)="Hans Mueller"                     [3006]
9320 z$(2)="5309 Meckenheim, "                [2136]
9330 z$(3)="Elserweg 12"                      [2099]
9340 REM                                      [272]
9380 PRINT#8,TAB(7);z$(1);:PRINT#8,TAB(45) [4000]
;z$(2);:PRINT#8,dat$
9390 PRINT#8,SPACE$(44);z$(3)                 [1564]
9400 RETURN                                   [555]
9500 REM **** Unterprogramm warten auf Tas [3028]
te *****
9510 CLS#3:LOCATE#3,30,2:PRINT#3,"Bitte Ta [4511]
ste druecken.........."
9520 CALL &BB06                               [393]
9530 CLS#3                                    [371]
9540 RETURN                                   [555]
```









"...sehr gut gelungenes Actionspiel mit Abenteuer-Touch und sehr hohem Spielwitz."
(HAPPY COMPUTER 07/86)

Mission ELEVATOR ist lieferbar für:
COMMODORE und
SCHNEIDER
(Kassette, Diskette)

MIT DEUTSCHER SPIELANLEITUNG!

Mission ELEVATOR

EURO GOLD
STARLINE

SCHNEIDER CPC
Diskette

Deutsch/Englisch

MICROPOOL Produkte erhalten Sie in den Fachabteilungen von Horten Horten Horten und Quelle INTERNATIONAL sowie in gutsortierten Computershops.

Mission ELEVATOR bestellen Sie bei Quelle unter der Kat.-Nr.
CPC-Kass 560-632-2
C64-Kass 560-813-8
C64-Disk 560-823-7

Vertrieb:
RUSH WARE
Online with the trend.

Mitvertrieb:
Microhändler GmbH